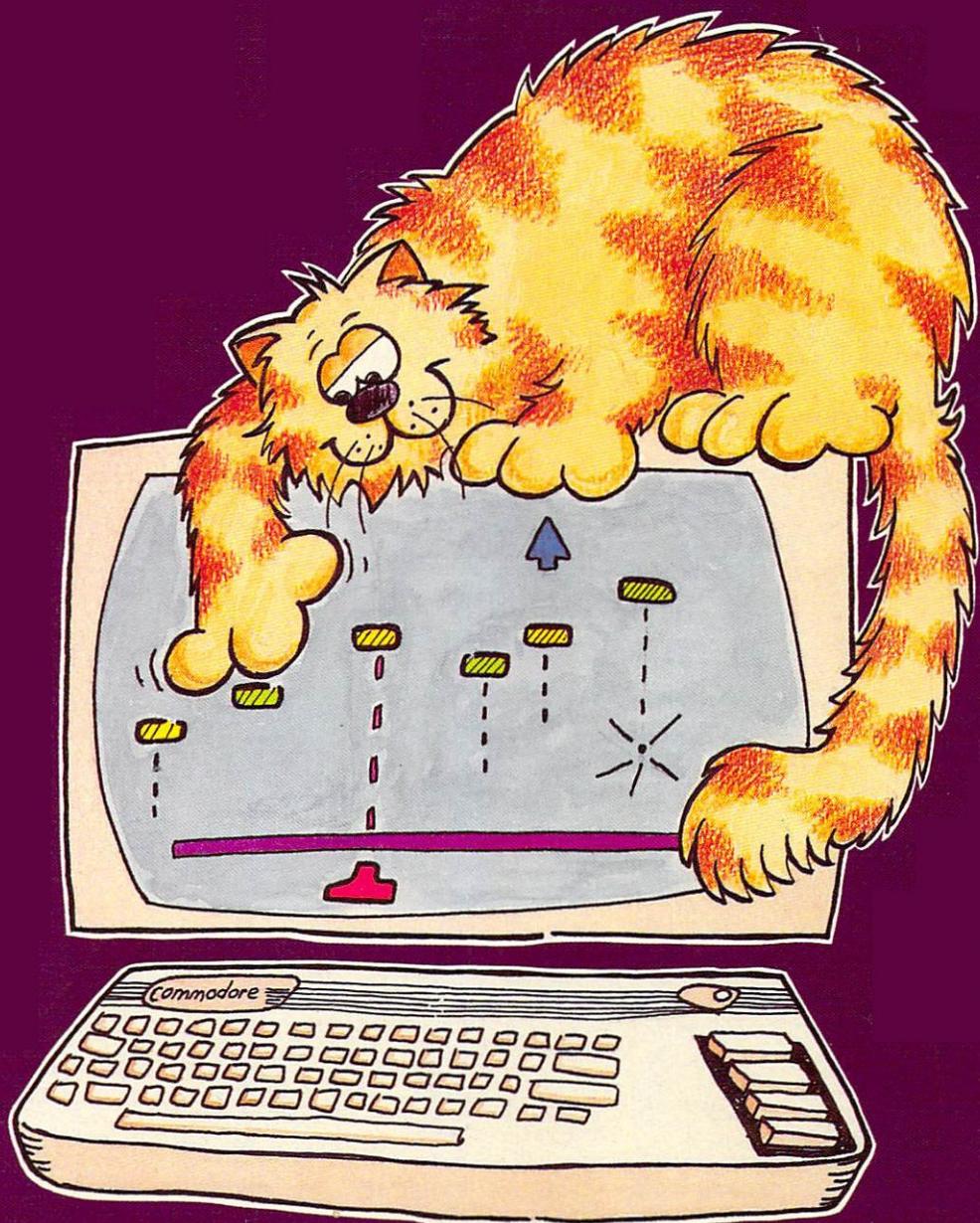


COMMÓDORE 64™

Fun and Games

Jeffries
Fisher
Sawyer



COMMODORE 64TM

FUN AND GAMES

**Ron Jeffries
Glen Fisher
Brian Sawyer**

**Osborne/McGraw-Hill
Berkeley, California**

Disclaimer of Warranties and Limitation of Liabilities

The authors have taken due care in preparing this book and the programs in it, including research, development, and testing to ascertain their effectiveness. The authors and the publisher make no expressed or implied warranty of any kind with regard to these programs nor the supplementary documentation in this book. In no event shall the authors or the publishers be liable for incidental or consequential damages in connection with or arising out of the furnishing, performance, or use of any of these programs.

COPYRIGHT. This collection of programs and their documentation is copyrighted. You may not copy or otherwise reproduce any part of any program in this collection or its documentation, except that you may load the programs into a computer as an essential step in executing the program on the computer. You may not transfer any part of any program in this collection electronically from one computer to another over a network. You may not distribute copies of any program or its documentation to others. Neither any program nor its documentation may be modified or translated without written permission from Osborne/McGraw-Hill.

NO WARRANTY OF PERFORMANCE. Osborne/McGraw-Hill does not and cannot warrant the performance or results that may be obtained by using any program in this book. Accordingly, the programs in this collection and their documentation are sold "as is" without warranty as to their performance, merchantability or fitness for any particular purpose. The entire risk as to the results and performance of each program in the collection is assumed by you. Should any program in this collection prove defective, you (and not Osborne/McGraw-Hill or its dealer) assume the entire cost of all necessary servicing, repair, or correction.

LIMITATION OF LIABILITY. Neither Osborne/McGraw-Hill nor anyone else who has been involved in the creation, production, or delivery of these programs shall be liable for any direct, incidental, or consequential benefits, such as, but not limited to, loss of anticipated profits or benefits, resulting from the use of any program in this collection or arising out of any breach of any warranty. Some states do allow the exclusion or limitation of direct incidental or consequential damages, so the above limitation may not apply to you.

Published by
Osborne/McGraw-Hill
2600 Tenth Street
Berkeley, California 94710
U.S.A.

**For information on translations and book
distributors outside
of the U.S.A., please write to
Osborne/McGraw-Hill at the above address.**

C-64 is a trademark of Commodore Business Machines, Inc. *C-64 Fun and Games* is not sponsored or approved by or connected with Commodore Business Machines, Inc. All references to C-64 in the text of this book are to the trademark of Commodore Business Machines, Inc.

CBM is a trademark of Commodore Business Machines, Inc.

CURSOR (TM) is a trademark of The Code Works.

Master Mind (R) is a registered trademark of Invitica Plastics.

Othello (R) is a registered trademark of Gabriel Industries, Inc.

Yahtzee (R) is a registered trademark of E.S. Lowe Company, Inc.

Chutes and Ladders (R) is a registered trademark of Milton Bradley Company.

COMMODORE 64™ FUN & GAMES

Copyright © 1984 by McGraw-Hill. All rights reserved. Printed in the United States of America. Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a data base or retrieval system, without the prior written permission of the publisher, with the exception that the program listings may be entered, stored, and executed in a computer system, but they may not be reproduced for publication.

1234567890 DODO 89876543

ISBN 0-88134-116-9

Judy Ziajka, Acquisitions Editor
Brad Hellman, Technical Editor
Elizabeth Swigert, Copy Editor
Sharon Smith, Cover Design
Jenni Jeffries, Cover and Text Illustrations

Contents

INTRODUCTION	v	
TINY-1	1	
TINY-2	2	
TINY-3	3	
TINY-4	5	
TINY-5	7	
Each Tiny is a short program that displays different patterns and colors on your screen.		
ROADHOG	9	
Test your driving skill along a narrow and crooked road.		
PIANO	12	
Perform catchy melodies with a screen keyboard playing along.		
BETS	17	
Try to outbet three opponents in this unusual card game.		
SAFE	23	
Can you burglarize a safe before being caught by the law?		
BOSWAIN	31	
Pay attention to this guy's hands because he has a hungry eye on yours.		
HANOI	38	
Figure out how to move the golden disks that some Asian priests have been moving for centuries.		
MISER	43	
Risk your life trying to discover hidden treasure in a haunted mansion.		
MAD	53	
The C-64 creates interesting stories from words that you supply.		
GODZILLA	58	
One of the world's most famous monsters is on the rampage again.		
RATRUN	69	
Wander through a maze from a "rat's eye" view in search of cheese.		
YAHTZEE	75	
If you think rolling five dice to come up with the same number is easy, perhaps you should try this game.		
LEAP	81	
Remove pegs one by one simply by leaping over them, but try to end up with just one peg.		
BOX	85	
Shoot rays into a box to discover hidden atoms.		
LAWN	91	
Mow a suburban lawn from the comfort of your chair, but you must finish before running out of gas.		
KALAH	95	
Sow pebbles into your kalah and you'll be a winner.		
BONZO	102	
Race to the top, but watch the obstacles that are thrown in.		
BJACK	107	
Deal another hand because this imaginary card game allows you all the money you can spend.		
FIRE	114	
The inferno has started and only you can prevent total disaster.		
ZAP	119	
Maneuver your "zapper" around an arena and score points by "zapping" your opponents.		
EVEREST	123	
The harsh elements of the world's tallest mountain hinder your trek to the peak.		
REVERSI	129	
The game may be as old as time itself, but just try to defeat the C-64 with strategy and guile.		
BOP	136	
If you think counting on your fingers is old-fashioned, try counting this way.		
SPOT	143	
You only need four squares in a row to win, but the fun is how you place the squares.		

DOTS	147	PIEGRAM	169
You may have thought you had played everyone in this game, but you haven't lived until you play the C-64.			
CAPTURE	152	BAT	174
Two beasts chase you and try to devour you while you try to capture them.			
DIVE	157	RESCUE	178
You command divers to retrieve sunken treasure.			
STOP	163	Your mission is to rescue comrades on an alien planet while attackers try to thwart your operation.	
Try to capture 3 out of 11 columns on a board before your opponent does.			

INTRODUCTION

Commodore 64 *Fun and Games* provides you with complete listings for 35 programs written specifically for the Commodore 64 home computer. (NOTE: These programs do not work on the Commodore VIC 20 or the Commodore PET.)

The programs use the special features the C-64 offers, such as color, sound, sprites, and character graphics. All of the programs are written in BASIC, but you don't need to be a programmer to play these games. All you need to do is enter them from the listings provided.

HOW TO USE THIS BOOK

A complete listing is provided for each program. You type the program into your computer exactly as it is listed in the book and then play. Instructions on how to play each game are given in the introduction to each program.

Whenever you enter a program, be sure to save it on cassette or diskette *before* you try to run it. If you've made a typing mistake that causes your C-64 to "lock up," you will be able to load the program from cassette or diskette and correct the error. This method of saving a program before running it may save you hours of retyping.

HOW TO READ THE LISTINGS

Most of the programs use C-64 special characters and functions, such as color keys, reverse video, cursor control, and character graphics. We use our own special notation in the program listings to make them easier to read. Before you enter any of the listings, take a few minutes to become familiar with the notation. Although it may look strange at first, once you learn the notation you will be able to type in the programs correctly.

We do not use listings produced by a Commodore printer for a good reason: it's difficult to type lengthy C-64 programs that use graphics when working from C-64 produced listings. The Commodore printer uses graphic symbols for the special functions, such as cursor movement or changing colors. When you type in a program from a C-64 produced listing, you have

to remember what the large set of graphic characters means. The notation used in this book indicates which keys to press when you type in the program. Look at Table 1 if you want to see what graphic character the C-64 will display when you type the appropriate key.

The key to entering programs successfully from this book of listings is to be careful. We've included several short programs called TINY-1 through TINY-5 to help you get started. The best way to learn the notation used in this book is to try typing several of these TINY programs.

If you hurry while typing in a program, you may later discover that the program doesn't work when you try to run it. When this happens, you'll have to compare each line you typed against the listing in the book. You'll quickly discover that slowing down and carefully entering the program will take much less time than "debugging" a program that was entered incorrectly.

The Listings

Most often you will type exactly what is printed in the listings. When you see the curly braces ("{" and "}"), you know that we are using our special notation, which will be explained below. Otherwise, type exactly what is shown in the listing, with one exception. It is often important that you type in a specified number of blank spaces. If a blank space is important, we will show it as the caret ("^") which is not a character on the C-64. Here is an example:

```
1 PG$="^P^I^A^N^O^":AU$="BY^GLEN^FISHER"
```

In this line from the PIANO program, there are eight blanks that are required. Since these blanks must be entered, they are represented by the caret symbol.

In C-64 BASIC, blanks are not always required in a program statement. Sometimes you use blanks to make the program easier to read. We've shown these "optional" blanks as actual blanks in the listings. Although the programs will work even if you leave

Table 1. Notation Used for Program Listings

Listed:	Key(s) to Press:	Screen Graphics:	Listed:	Key(s) to Press:	Screen Graphics:
{home}	HOME		{blu}	COMMODORE 7	
{clr}	SHIFT CLR		{yel}	COMMODORE 8	
{up}	SHIFT CRSR UP		{rvs-on}	CTRL 9	
{down}	CRSR DOWN		{rvs-off}	CTRL 0	
{left}	SHIFT CRSR LEFT		{pi}	SHIFT ↑	
{right}	CRSR RIGHT		{space}	SPACE BAR	
{inst}	SHIFT INST		{shift space}	SHIFT SPACE BAR	
{blk}	CTRL 1		{A}	SHIFT A	
{wht}	CTRL 2		{B}	SHIFT B	
{red}	CTRL 3		{C}	SHIFT C	
{cyn}	CTRL 4		{D}	SHIFT D	
{pur}	CTRL 5		{E}	SHIFT E	
{grn}	CTRL 6		{F}	SHIFT F	
{blu}	CTRL 7		{G}	SHIFT G	
{yel}	CTRL 8		{H}	SHIFT H	
{blk}	COMMODORE 1		{I}	SHIFT I	
{wht}	COMMODORE 2		{J}	SHIFT J	
{red}	COMMODORE 3		{K}	SHIFT K	
{cyn}	COMMODORE 4		{L}	SHIFT L	
{pur}	COMMODORE 5		{M}	SHIFT M	
{grn}	COMMODORE 6		{N}	SHIFT N	

Table 1. Notation Used for Program Listings (*continued*)

Listed:	Key(s) to Press:	Screen Graphics:	Listed:	Key(s) to Press:	Screen Graphics:
{O}	SHIFT O		{J}	COMMODORE J	
{P}	SHIFT P		{K}	COMMODORE K	
{Q}	SHIFT Q		{L}	COMMODORE L	
{R}	SHIFT R		{M}	COMMODORE M	
{S}	SHIFT S		{N}	COMMODORE N	
{T}	SHIFT T		{O}	COMMODORE O	
{U}	SHIFT U		{P}	COMMODORE P	
{V}	SHIFT V		{Q}	COMMODORE Q	
{W}	SHIFT W		{R}	COMMODORE R	
{X}	SHIFT X		{S}	COMMODORE S	
{Y}	SHIFT Y		{T}	COMMODORE T	
{Z}	SHIFT Z		{U}	COMMODORE U	
{A}	COMMODORE A		{V}	COMMODORE V	
{B}	COMMODORE B		{W}	COMMODORE W	
{C}	COMMODORE C		{X}	COMMODORE X	
{D}	COMMODORE D		{Y}	COMMODORE Y	
{E}	COMMODORE E		{Z}	COMMODORE Z	
{F}	COMMODORE F		{+}	SHIFT +	
{G}	COMMODORE G		{±}	COMMODORE +	
{H}	COMMODORE H		{-}	SHIFT -	
{I}	COMMODORE I		{=}	COMMODORE -	

Table 1. Notation Used for Program Listings (*continued*)

Listed:	Key(s) to Press:	Screen Graphics:	Listed:	Key(s) to Press:	Screen Graphics:
{ * }	SHIFT *		{ \ }	COMMODORE \	
{ _ }	COMMODORE *		{ @ }	SHIFT @	
{ \ }	SHIFT \		{ @ }	COMMODORE @	

out the optional blanks, we strongly recommend that you use them.

To make the program listings conform to the format of this book, we "folded" long statement lines at a convenient place on the line and continued the statement on the next line. When you type the program in, you should ignore where we have "wrapped" the line around and just keep typing. When long statement lines are continued, they are aligned with the rest of the text. Since the screen of the C-64 is 40 columns wide, the longer lines of the programs will look different on your screen than they do in the book.

Another minor point about these listings is that we show the line numbers aligned to the right, rather than the left as they will appear on the C-64 screen. This was done to make the listings look more attractive and easier to type in.

Special Notation

Everything inside curly braces ("{" and "}") is our special notation that tells you which keys to press on the C-64 keyboard. There are three types of keys that will appear within the braces: keys that move the cursor, keys that control color, and keys that produce C-64 graphic symbols.

There are several keys that we've given special names. For example, we use {home} to mean "press the HOME key." This is the easiest part of the notation to learn, since all you need to know is what we've decided to call the various keys. Table 1 has three different headings: what will appear in our listing, the actual key you need to press, and a picture of the graphic character that will appear on the screen.

When you need to enter a "clear screen" code in a program, we will show {clr} in the listing. As shown in

Table 1, you will hold down the SHIFT key and press the CLR key. A reverse-video heart graphic character will appear on the screen.

It is useful to notice that the keys we have named, such as {clr}, are all printed in lowercase letters. When capital letters appear within the braces, they always represent graphic characters produced with either the SHIFT or the COMMODORE key. Since the keys we have named are all printed in lowercase, there isn't any possibility of confusion between names of keys and the same set of keys used to produce graphics.

Cursor Movement Keys

To indicate cursor movement keys that are entered in strings, we use {home}, {clr}, {up}, {down}, {left}, and {right}. The "insert character" function is shown as {inst}.

Color Keys and Reverse-Video

As you know, the C-64 can display 16 colors. Abbreviations for eight of the colors are printed on the front of the number keys. To produce these colors, you hold down the CTRL key and press the number key with the name of the color you want. With each listing, we show these first eight colors by printing the name of the color on the front of the key: {blk}, {wht}, {red}, {cyn}, {pur}, {grn}, {blu}, and {yel}. These colors are names for the keys, just as {down} is the name for the cursor-down key.

The remaining eight colors, whose names are not printed on the C-64 keycaps, are shown as the *underlined names* of the colors which are on the keycaps. When you see an underlined color name, it means "hold down the COMMODORE key and press the key

with the appropriate color name." For example, {blk} is how we indicate orange, the second color for the 1 key. You will see below that underlining within braces *always* means "hold down the COMMODORE key."

To turn reverse-video on and off, we use {rvs-on} and {rvs-off}. By looking at Table 1, you can see that this involves pushing two keys. You can produce the reverse-video by holding the CTRL key down while pressing 9 for {rvs-on} (reverse on) or 0 for {rvs-off} (reverse off).

A Simple Example

Here is an example of a PRINT statement that clears the screen, uses a cursor-right to move one space to the right, goes down one row, turns on the color red, and prints "HI" on the screen:

10 PRINT "{clr right down red} HI"

To enter the example, type the following:

- the number 10
- the word PRINT
- a quote ("")
- the CLR key
- cursor-right key
- cursor-down key
- hold down the CTRL key and press the number 3
(for the color red)
- the letters H, I
- and the closing quote (")

Note that blanks *inside* curly braces are never typed unless the word {space} is spelled out. Blanks are used to separate the names of special keys and are included for clarity only. For example, {down down left} means press three keys: cursor-down, cursor-down, and cursor-left. The blanks between the names are *not* entered into the computer.

Graphics that Use the SHIFT Key

Most keys on the C-64 keyboard can produce three different codes: a letter, such as an "X", and two different graphic characters. The graphic character shown on the right front of a key is produced by holding the SHIFT key down and pressing the key with the desired graphic. For example, to produce the heart character, hold down the SHIFT key and then press the S key. To produce the graphic character

shown on the left front of a key, hold the COMMODORE key down and press the key with the graphic character.

The graphics produced with the SHIFT key are shown as capital letters inside the braces. For example, {X} is how we show "hold down SHIFT, and press the letter X." Table 1 shows all of the graphics produced by these shifted keys.

The plus and minus sign, the commercial "at" sign, the asterisk, and the British "pound" sign are other graphics produced by the SHIFT key when they appear inside braces.

Graphics That Use the COMMODORE Key

We show the COMMODORE key graphics just like the SHIFT graphic characters, except we *underline* the key you should press. For example, {X} will produce the character Ⓜ. Underlined symbols give you the graphic character shown to the left on the front of the C-64 keycap, while the non-underlined symbols produce the graphic shown on the right side of the keycap.

Repeated Cursor Movements and Spaces

Some cursor movements involve a lot of repetition, for example, {home down down down down}. To save space and increase clarity, we show such repetitions by a key count followed by the key name. The example above would be shown as:

{home 4° down}

Here's an example that shows how we print a home, 10 cursor-downs, and 20 cursor-rights, followed by "HELLO" (printed in yellow):

10 PRINT "{home 10° down 20° right yel} HELLO"

Since blank spaces tend to be used in long strings, we also use the repetition factor when more than three important blanks appear together. This makes it far easier to type in the right number of spaces.

JOYSTICKS AND SOUND

Many of the games use the C-64 joystick. For such games, plug the joystick into control port 2. Port 2 is used because port 1 can sometimes interfere with the keyboard.

When you use the joystick, hold it so that the "fire button" is at the upper left-hand corner. We've found

that the joystick can become unplugged quite easily. To prevent this, use a cassette case to support the joystick plug next to the port. The case is just the right height to relieve strain on the joystick plug and help it operate reliably.

Many of the games use the C-64 sound capability. You will want to adjust the volume of the sound on your television or monitor so that you can hear it.

THE STANDARD FRAMEWORK

To save you effort in typing, all programs except the TINY series use a "standard framework" that starts at line 60000. In the program ROADHOG, the standard framework is shown with a gray background on page 11. (There are two exceptions: the BETS program has an added line at 60025, and GODZILLA has an added line at 62035.) The standard framework puts the title on the screen, handles the joystick control, provides a keyboard-handling routine, and sets several variables that are used by most programs.

You can save a lot of typing by carefully entering the standard framework by itself and saving it on cassette or diskette. Whenever you are ready to type in a new program, first load the standard framework. If you do this, you'll be able to stop typing when you reach line 60000, since the standard portion of the program will already be there. (Don't forget that BETS and GODZILLA need an extra line, though!)

You can terminate the execution of any program by typing a Q.

COMMON MISTAKES TO AVOID

There are several mistakes that you can avoid if you know about them in advance. It is easy to mistake

certain letters and numbers that are similar. The letter "I" and the number "1" are often confused, as are the letter "O" and the number "0" (zero).

Remember that underlined capital letters inside the braces use the COMMODORE key, while those that are not underlined use the SHIFT key. Don't forget that blank spaces inside the braces are there purely for smooth reading. When you are required to type a space, it will be a caret (^) outside of the braces or the word "space" inside the braces.

When cursor control keys are involved, make sure that you type the correct ones. If you accidentally enter a cursor-right where the program listing shows {left}, you'll find that the program won't work correctly! And when there are many cursor movement keys or spaces together, be sure that you type the correct number of keys. One too many or one too few characters in a string can create havoc with a program.

When you enter DATA lines, be especially careful. DATA statements are often long sequences of numbers or letters that are fairly easy to type incorrectly.

OBTAINING PROGRAMS ON DISKETTE

All programs are available on diskette (although you will need this book for instructions and the program listings). For more information, you should write to:

Commodore-64 Fun and Games
c/o The Code Works
Box 6905
Santa Barbara, CA 93160
U.S.A.

TINY-1

By Brian Sawyer



This is a tiny program for the Commodore 64 that you should be able to type in less than ten minutes. TINY-1 is a simple graphics demonstration that paints a pleasing pattern of color on your monitor or television screen. After you have completed typing in the program, check it carefully against the program listing; then save it on a cassette tape or a floppy disk before you run the program. (This is an important habit to form: always save the program before running it. Sometimes an error made

in entering a program will cause the C-64 to "lock up" when you attempt to run it. If you haven't saved the program, you could lose hours of typing!) The program will continue to change the colored patterns until you press the letter Q, however, there will be a slight delay before the program actually halts.

Unlike almost all of the other programs in this book, TINY-1 does not need our "standard framework" in order to run. We've done this to make it easy for you to get started.

```
100 CM=55296:WD=40:VIC=53248
105 W=W+1:BL$="{rvs-on 39° space 2°left inst}\"
110 POKE VIC+32,0:POKE VIC+33,0:PRINT "{clr blk}";:FOR N=1 TO 24:
    PRINT BL$
120 NEXT:PRINT BL$ "{home}"
150 FOR Q=12 TO 1 STEP -1:GOSUB 200:NEXT
160 FOR Q=1 TO 12:GOSUB 200:NEXT:FOR Z=1 TO 100:NEXT Z:GOTO 150
200 UL=CM+Q+Q*WD:UR=CM+Q*WD+(39-Q):LL=CM+(25-Q)*WD+Q:
    LR=CM+(39-Q)+WD*(25-Q)
205 W=W+1:IF W>15 THEN W=0
210 FOR N=UL+1 TO UR:POKE N,W:NEXT
220 FOR N=UR TO LR STEP WD:POKE N,W:NEXT
230 FOR N=LR-1 TO LL STEP -1:POKE N,W:NEXT
240 FOR N=LL TO UL STEP -WD:POKE N,W:NEXT
250 GET Z$:IF Z$<>"Q" THEN RETURN
260 PRINT "{clr blu}":END
```

TINY-2

By Glen Fisher



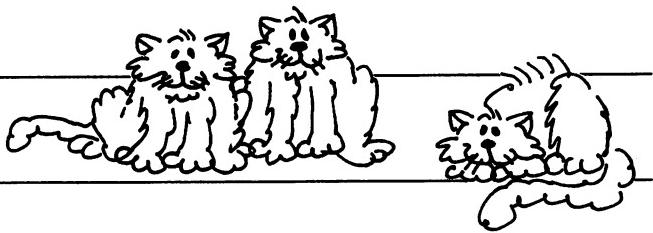
Here's another program that should take less than ten minutes to type in and run correctly. TINY-2 produces an attractive multi-

colored design on the screen of the C-64. Let it run for a few minutes, and you'll be treated to some inexpensive modern art.

```
100 VIC=53248:FOR I=1 TO 4:FOR J=1 TO 3:READ SY$(I,J),ND(I,J):
      NEXT J,I
110 FOR I=1 TO 4:READ RC(I),CC(I):NEXT I
120 D=INT(RND(1)*4+1)
130 R=12:C=20:PRINT "{clr 12°down}";TAB(C);"+ left";:
      POKE VIC+33,0
140 R=R+RC(D):C=C+CC(D):N=6
150 T=INT(RND(1)*N+1):IF T>3 THEN T=3
160 ND=ND(D,T):R2=R+RC(ND):C2=C+CC(ND)
170 IF R2<1 OR R2>23 OR C2<1 OR C2>38 THEN 150
180 IF RND(1)>.96 THEN PRINT LEFT$("{cyn pur grn yel blk}",
      RND(1)*5+1);
190 PRINT MID$("{up right down left}",D,1);SY$(D,T);{left};
200 GET A$:IF A$="Q" THEN END
210 D=ND:R=R2:C=C2:GOTO 150
260 DATA "{A}",2,"{S}",4,"{-}",1
270 DATA "{S}",3,"{X}",1,"{*}",2
280 DATA "{X}",4,"{Z}",2,"{-}",3
290 DATA "{Z}",1,"{A}",3,"{*}",4
300 DATA -1,0,0,1,1,0,0,-1
```

TINY-3

By Glen Fisher



This little program creates symmetrical figures on the screen of the C-64. When you run TINY-3, there will be a short delay while it calculates the first picture. After the design is displayed, the program will calculate another design.

```
1000 VIC=53248:POKE VIC+33,0:MN=20:MX=100:DIM P(MX+1),H(4),V(4),
      RG(6),R(6)
1005 PRINT "{clr 2°down wht 2°right}LOADING..."
1010 HL=4:HH=2:VL=3:VH=1:HC=5:VC=6
1020 LF=1:AH=2:RT=3:RV=2
1100 FOR I=1 TO 4:FOR J=1 TO 3:READ D(I,J),D$(I,J):NEXT J,I
1110 FOR I=1 TO 4:READ H(I),V(I):NEXT I
2000 N=INT(RND(1)*(MX-MN+1))+MN
2010 H=0:V=0:D=VH
2020 FOR I=1 TO 4:R(I)=0:RG(I)=0:NEXT I
2100 FOR I=1 TO N
2110 P=INT(RND(1)*3)+1
2120 GOSUB 8400
2130 P(I)=P
2135 GET T$:IF T$="Q" THEN END
2140 NEXT I
2150 IF D=VH THEN N=N+1:P=LF:GOSUB 8400:P(N)=P
2160 IF D=HL THEN TN=LF
2170 IF D=VL THEN TN=RV
2180 IF D=HH THEN TN=RT
2200 FOR I=0 TO 4:R(I)=RG(I):RG(I)=0:NEXT I
2210 R(HC)=H:R(VC)=V
2220 H=0:V=0:D=VH
2300 GOSUB 8300
2310 H=H+R(HC):V=V+R(VC)
2320 ON TN GOSUB 8000,8100,8200
2330 IF D<>VH THEN 2300
2400 IF RG(HH)-RG(HL)>72 THEN 2000
2410 IF RG(VH)-RG(VL)>46 THEN 2000
3000 HO=(72-(RG(HH)-RG(HL)))/2-RG(HL)
3005 HO=INT(HO/2)+3
3010 VO=(46-(RG(VH)-RG(VL)))/2+RG(VH)
3015 VO=INT(VO/2)
3020 PRINT "{clr}"
3080 PRINT LEFT$(" {home 25°down}",VO+1);TAB(HO);:Q=POS(0)
```

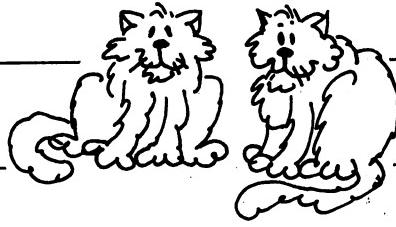
```

3090 D=VH
3100 FOR I=1 TO N
3110 PRINT "{Q left}";D$(D,P(I));:D=D(D,P(I))
3120 GET T$:IF T$<>"" THEN 3200
3150 NEXT I
3160 IF D<>VH THEN 3100
3170 GOTO 2000
3200 RETURN
8000 REM
8010 T=R(VH):R(VH)=R(HH):R(HH)==R(VL):R(VL)=R(HL):R(HL)==T
8020 T=R(HC):R(HC)==R(VC):R(VC)=T
8030 D=D(D,LF)
8040 RETURN
8100 REM
8110 T=R(HH):R(HH)==R(HL):R(HL)==T
8120 T=R(VH):R(VH)==R(VL):R(VL)==T
8130 R(HC)==R(HC):R(VC)==R(VC)
8140 D=D(D,RT),RT
8150 RETURN
8200 REM
8210 T=R(VH):R(VH)==R(HL):R(HL)=R(VL):R(VL)==R(HH):R(HH)=T
8220 T=R(HC):R(HC)=R(VC):R(VC)==T
8230 D=D(D,RT)
8240 RETURN
8300 T=R(HL)+H:IF T<RG(HL) THEN RG(HL)=T
8310 T=R(HH)+H:IF T>RG(HH) THEN RG(HH)=T
8320 T=R(VL)+V:IF T<RG(VL) THEN RG(VL)=T
8330 T=R(VH)+V:IF T>RG(VH) THEN RG(VH)=T
8340 RETURN
8400 ND=D(D,P):H=H+H(D)+H(ND):V=V+V(D)+V(ND)
8410 GOSUB 8300:D=ND
8420 RETURN
9000 DATA 4,"{I 2°left}",1,"{- left up}",2,"{U}"
9010 DATA 1,"{K left up}",2,"{*}",3,"{I left down}"
9020 DATA 2,"{J}",3,"{- left down}",4,"{K 2°left}"
9030 DATA 3,"{U left down}",4,"{* 2°left}",1,"{J left up}"
9040 DATA 0,1,1,0,0,-1,-1,0

```

TINY-4

By Brian Sawyer



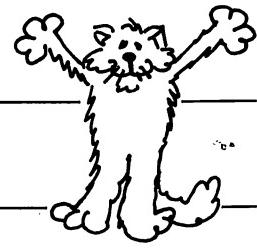
This program draws a picture of a volcano which then erupts with lava and large hunks of rock.

```
1000 VIC=53248:CRT=1024:WD=40:POKE VIC+33,0:GOSUB 1600
1001 DI(1)=WD-1:DI(2)=WD:DI(3)=WD+1:DI(4)=-1:DI(6)=1:DI(7)=-WD-1:
    DI(8)=-WD
1005 DI(9)=-WD+1
1015 DIM A(50):FOR N=1 TO 30:IF N<26 THEN READ Q:A(N)=DI(Q)
1020 A(N)=A(N)+A(N-1):NEXT
1025 DIM B(50):FOR N=1 TO 30:IF N<26 THEN READ Q:B(N)=DI(Q)
1030 B(N)=B(N)+B(N-1):NEXT
1040 TP=CRT-WD*3-7
1050 F=81:G=32
1060 DIM D$(21):FOR N=1 TO 20 STEP 2:D$(N)="{rvs-on red £ blk £ red
    2°* rvs-off}":D$(N+1)="{rvs-on red * blk * yel £ red £}":NEXT
1070 B$="{home 12°down 15°right}":E$="{2°up right}..{down 2°left}.."
1080 A$="{2°up right rvs-on yel * red £ down 2°left N blk M down
    3°left * yel M red N£ rvs-off}"
1090 N1=TP+WD*13+25:N2=TP+WD*13+23:N3=TP+WD*13+24:N4=TP+WD*12+24:
    N5=TP+WD*13+22
1100 N6=TP+WD*12+23:N7=TP+WD*12+22:N8=TP+WD*12+25
1110 GOSUB 1320
1120 K(1)=205:K(2)=206
1130 FOR N=1 TO 1200:NEXT
1140 GOTO 1210
1150 POKE N1+A(Q1-1),G:POKE N2+A(Q2-1),G:POKE N3+A(Q3-1),G:
    POKE N7+A(Q7-1),G
1160 POKE N4+B(Q4-1),G:POKE N5+B(Q5-1),G:POKE N6+B(Q6-1),G:
    POKE N8+B(Q8-1),G
1170 IF Q3=17 THEN 1210
1180 POKE N1+A(Q1),F:POKE N4+B(Q4),F:POKE N2+A(Q2),F
1185 POKE N7+A(Q7),F:POKE N8+B(Q8),F:POKE N5+B(Q5),F
1190 POKE N3+A(Q3),F:POKE N6+B(Q6),F:F=46:PRINT B$D$(Q3)
1195 GET IN$:IF IN$="Q" THEN PRINT "{blu}":END
1200 Q1=Q1+1:Q2=Q2+1:Q3=Q3+1:Q4=Q4+1:Q5=Q5+1:Q6=Q6+1:Q7=Q7+1:
    Q8=Q8+1:GOTO 1150
1210 F=81:PRINT B$A$
1220 Q2=INT(RND(10)*5)
1230 Q3=INT(RND(10)*5)
1240 Q4=INT(RND(10)*5)
```

```

1250 Q5=INT(RND(10)*5)
1260 Q6=INT(RND(10)*5)
1270 Q7=INT(RND(10)*5)
1280 Q8=INT(RND(10)*5)
1290 Q1=INT(RND(10)*5)
1300 PRINT B$E$
1310 GOTO 1180
1320 PRINT "{home 10°down wht}"
1330 PRINT "{16°right red rvs-on £*}"
1340 PRINT "{15°right rvs-on £}~{*}"
1350 PRINT "{14°right rvs-on £ 4°space _*}"
1360 PRINT "{13°right pur N red *£* rvs-on}~{rvs-off £ pur M}"
1370 PRINT "{12°right N 4°space red *£ pur}~{M}"
1380 PRINT "{11°right N 10°space M}"
1390 PRINT "{10°right N}~{shift-space 10°space M}"
1400 PRINT "{9°right N}~{shift-space 12°space M 5°space NM}"
1410 PRINT "{8°right N}~{shift-space 14°space M}~{N}~{M}"
1420 PRINT "{5°right NMN}~{grn}↑{5°space}↑{5°space}↑~↑{pur M}~{N
4°space M}"
1430 PRINT "{4°right N}~{N grn}↑↑~↑↑{10°space}↑~↑{shift-space}
↑↑{6°space pur M}"
1440 PRINT "{3°right grn}↑~↑↑~↑~↑↑{6°space}↑~↑~↑~↑↑{6°space pur
M}"
1450 RETURN
1500 DATA 8,8,8,9,9,6,6,6,3,3,3,3,3,3,2,2,2,2,2,2,2,2,2,2,2,2,2,2
1510 DATA 8,8,8,7,7,4,4,4,1,1,1,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2
1600 T=PEEK(VIC+17):POKE VIC+17,0
1610 POKE VIC+33,1:PRINT "{clr}"
1620 POKE VIC+33,0:POKE VIC+17,T
1630 RETURN

```



TINY-5

By Brian Sawyer

Just what you always wanted: a kaleidoscope for your C-64.

```
100 CRT=1024:HT=24:WD=40:PRINT "{clr}"
110 VIC=53248:POKE VIC+33,0
120 WIDTH=38:X=(WD-4-WI)/2+2:REM +3
130 HEIGHT=23:Y=(HT-HE)/2+1
140 C$="{home 25°down}"
150 DIM CH$(22)
160 FOR J=0 TO 22:READ CH$(J):NEXT J
170 C1=1:C2=2:C3=3:C5=5:C7=7
180 RX=INT(RND(1)*WI/2)
190 RY=INT(RND(2)*HE/2+1)
200 CH$=CH$(INT(RND(1)*23))
210 PRINT LEFT$("{wht red cyn pur grn blu yel blk}",RND(1)*8+1);
220 PRINT LEFT$(C$,Y+RY);TAB(X+RX);MID$(CH$,C1,C2);
230 PRINT TAB(X+WI-RX-1);MID$(CH$,C3,C2)
240 PRINT LEFT$(C$,Y+HE-RY+1);TAB(X+RX);MID$(CH$,C5,C2);
250 PRINT TAB(X+WI-RX-1);MID$(CH$,C7,C2)
260 FOR L=1 TO 10*S
270 GET S$:IF S$="Q" THEN END
280 IF S$="" THEN NEXT L
290 L=VAL(S$):IF L>0 OR S$="0" THEN S=9-L
300 GOTO 180
310 DATA "{rvs-off K rvs-on K rvs-off K rvs-on K}","{rvs-on K
   rvs-off K rvs-on K rvs-off K}"
320 DATA "{rvs-off I rvs-off I rvs-on I rvs-on I}","{rvs-on I
   rvs-on I rvs-off I rvs-off I}"
330 DATA "{rvs-off £ rvs-off * rvs-on * rvs-on £}","{rvs-on £
   rvs-on * rvs-off * rvs-off £}"
340 DATA "{rvs-off * rvs-off £ rvs-on £ rvs-on *}","{rvs-on *
   rvs-on £ rvs-off £ rvs-off *}"
350 DATA "{rvs-off D rvs-off F rvs-off C rvs-off V}","{rvs-on D
   rvs-on F rvs-on C rvs-on V}"
360 DATA "{rvs-off C rvs-off V rvs-off D rvs-off F}","{rvs-on C
   rvs-on V rvs-on D rvs-on F}"
370 DATA "{rvs-off F rvs-off D rvs-off V rvs-off C}","{rvs-on F
   rvs-on D rvs-on V rvs-on C}"
380 DATA "{rvs-off V rvs-off C rvs-off F rvs-off D}","{rvs-on V
   rvs-on C rvs-on F rvs-on D}"
```

```
390 DATA "{rvs-off B rvs-on B rvs-on B rvs-off B}","{rvs-on B  
rvs-off B rvs-off B rvs-on B}"  
400 DATA "{rvs-off shift-space rvs-off shift-space rvs-off  
shift-space rvs-off shift-space}","{rvs-on}^{rvs-on}^{rvs-on}  
^{rvs-on}^"  
410 DATA "{rvs-off shift-space rvs-off shift-space rvs-off  
shift-space rvs-off shift-space}","{rvs-off shift-space  
rvs-off shift-space rvs-off shift-space rvs-off shift-space}"  
420 DATA "{rvs-off shift-space rvs-off shift-space rvs-off  
shift-space rvs-off shift-space}"
```

ROADHOG (Uses joystick)

By Brian Sawyer



ROADHOG is a simple game in which you control a race car using a joystick. As you can see, the program only takes about two and a half screens when you enter it into the computer. As we mentioned in the Introduction, you will need to create the "standard framework" routine for your C-64 and then save it on cassette tape or disk.

To enter the ROADHOG program, first load the standard framework into the computer, and then type in the program lines below. When you have finished typing in this program, clear the screen and give the command **LIST -5100**. Check each line carefully, making certain that you've entered everything correctly. When everything looks right, be sure to save the program on disk or cassette tape before doing anything else.

Now plug the joystick into the control port 2. Type

RUN and press RETURN. If all is well, you'll be controlling a small race car as it travels along a track that twists and turns and gradually gets narrower. If you are a good driver, you'll be able to navigate the full course successfully. If you crash by running off the track, a message will appear on the screen inquiring whether you wish to continue or halt. Just move the cursor with the joystick to your desired response, and then push the joystick button.

The car is represented by a "sprite" graphic; if you are familiar with sprite graphics on the C-64, you may want to modify the code to change the dimensions of the car.

Starting at line 60000 we show the standard framework in gray. You can save a lot of time by typing it in first and saving a copy on disk or cassette. See "Introduction" for details.

```
1 PG$="~R~O~A~D~H~O~G~":"AU$="BY~BRIAN~SAWYER":JF=1
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/19/83 BS
90 GOTO 62000
100 DIM L$(9),L(3),B(25),P%(1000):GOSUB 1000:POKE 53280,0:
    POKE 53281,0
110 POKE VIC,0:POKE VIC+23,0:POKE VIC+29,0:POKE VIC+27,000
115 N=19:PRINT "{clr 23°down wht}":V=-46:W=0
120 Q=35:M=2:K=162:U=0:I=0:B=0
121 POKE 2040,13:POKE 53287,8
125 POKE VIC,K:POKE VIC+1,160
128 L$(1)="{rvs-on £ 4°space}/{4°space rvs-off £}":
    L$(3)="{left _ rvs-on 4°space M 4°space _}":
    L$(2)="{rvs-on 4°space M 5°space}"
129 L$(4)="{rvs-on £}~~/{rvs-off £}":L$(6)="{left _ rvs-on}~~{M}~~{*}":L$(5)="{rvs-on}~~{M 4°space}~~{*}":L$(7)="{rvs-on £}~~/{rvs-off £}":L$(9)="{left _ rvs-on}~~{M}~~{*}":L$(8)="{rvs-on}~~{M}~~{*}"
```

```

132 PRINT "{clr}":FOR N=1 TO 25:PRINT TAB(11)L$(2):NEXT:
    PRINT "{wht}";:N=11
133 POKE VIC+21,1
135 L(1)=-1:L(2)=0:L(3)=1
140 PRINT TAB(N)L$(M+I)
142 U=U+1:IF U=110 THEN 300
145 POKE 53264,SGN(K AND 256):POKE 53248,K AND 255
155 Q=Q+1:A=PEEK(56320):IF A=123 THEN K=K-8
156 IF PEEK(1622+K/8)=32 THEN 400
157 GET A$:IF A$="Q" THEN 600
160 IF A=119 THEN K=K+8
175 IF RND(1)>.7 THEN M2=M:M=INT(RND(1)*3)+1
185 N1=N+L(M)
190 IF N1>27 OR N1<4 THEN N1=N:M=2
195 N=N1:GOTO 140
200 IF W=0 THEN W=1:V=-22:GOTO 210
205 W=0:V=-46
210 POKE 53287,W:POKE 53287,W:RETURN
300 IF B THEN 500
302 IF I=6 THEN B=B+1:PRINT "{grn}";:U=100:GOTO 145
305 IF M<>2 THEN U=109:GOTO 145
310 N=N+1:U=0:M=2:I=I+3
320 GOTO 145
400 FOR N=1 TO 30:POKE 53280,1:FOR T=1 TO 10:NEXT:POKE 53280,0:NEXT
410 POKE 53269,0:PRINT "{clr 5°down}YOU^CRASHED!":GOTO 545
500 FOR N=1 TO 1000:NEXT:POKE 53248+21,0
505 PRINT "{clr 5°down wht}YOU^FINISHED^THE^COURSE!"
545 PRINT:GOSUB 9100:PRINT "{wht}";:IF YN THEN 120
546 PRINT "{clr 2°down}":END
550 TM=TI+10
560 IF TM>TI THEN 560
570 RETURN
600 PRINT "{clr 3°down}THANKS^FOR^PLAYING!":POKE 53248+21,0:END
900 PRINT PEEK(56320):GOTO 900
1000 READ A:FOR N=64*13 TO 64*14-3:READ A:POKE N,A:NEXT:RETURN
9100 NO=0
9110 TM=0:SW=1
9115 PRINT "{wht}PLAY^AGAIN?^{\u00c3\u00b3}YES^NO{\u00c3\u00b3}^"
9120 IF TI<TM THEN 9150
9130 PRINT TAB(12+NO*4);MID$("{wht \u00c3\u00b3}",SW,1);
    MID$("YESNO{\u00c3\u00b3shift-space}",3*NO+1,3);"\u00c3\u00b3"
9140 SW=3-SW:TM=TI+15
9150 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
9160 IF JB AND JD=15 THEN YN=(NO=0):RETURN
9170 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 9115
9180 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 9115
9190 GOTO 9120

```

```

50000 DATA 62,0,0,0,71,255,226,239,60,247,254,24,127,253,255,191,237,
      193,183
50001 DATA 237,128,183,205,255,243,13,128,112,15,255,240,15,255,240,
      15,255,240
50002 DATA 15,255,240,239,255,240,255,255,247,255,255,255,255,255,
      255,247,255
50003 DATA 127,243,190,239,1,255,128,0,127
60000 IN$="^":ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MIDS("^.{+}",ZC,1);:{left}";:ZC=3-ZC:
      ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT ".{left}";:
      GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
      GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
      GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
      CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
      TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("21°space",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
      TAB(TB)LEFT$("21°space",LN)
62030 PRINT "2°down" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
      MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "4°down cyn" TAB(TB)MS$:PRINT "7°down cyn 9°space"
      (C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("pur cyn",K+1)LEFT$(MS$,Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

Note: The gray area is the standard framework.

PIANO

By Glen Fisher



This program displays a piano keyboard on the screen and then shows a bouncing dot as it plays one of three songs. You don't have to do anything to use PIANO except listen and watch as it performs some catchy melodies.

After you have the program working correctly, you may want to add new music or change the existing songs. The music for PIANO is stored in DATA statements using a notation that was designed to be compact and quickly processed. Unfortunately, these design goals precluded making the notation easy for a person to remember and enter into the computer. Keep in mind that this arbitrary code has nothing in common with conventional musical notation; in particular, the letters mean different things in the two notations.

In PIANO, notes are lettered from A to X; rests become an at-sign "@". The letters are arbitrary codes, not the names of the notes. Each letter is one halftone different from the letter on either side of it (for example, A is a halftone away from B, B is a halftone away from C, and so on).

Each note is followed by a number telling how many sixteenth notes it should play for. (Eighth notes play for two sixteenth notes, quarter notes for four, half notes for eight, and whole notes for sixteen.) To encode a piece of music, type the appropriate letters and numbers.

The data can also contain two quotes together forming an empty or "null" string, which is used for various special instructions; the exact instruction given is determined by the length of the note.

The following lists instructions and format used within the DATA statements that will control and play the music:

""0

Start playing all the music over again. It causes a RESTORE statement to be executed.

"",-1

Marks the start of a repeated passage of music.

"",-2,"Title",color

Marks the start of a song. "Title" is the name of the tune, to be printed on the screen. Color controls the background color during the piece. It should be the code number for the color you want.

","",n

Where n is a positive number, meaning that the preceding passage should be played n times. The "preceding passage" is everything back to the nearest "",-1. If no "",-1 appears in the song, everything back to the start of the song is repeated.

The general format for a song is as follows:

DATA "",-2, "scale",2

Marks the start of "scale", with a background color of red.

DATA G,2,I,2,K,2,L,2,N,2,P,2,R,2,S,2

The basic scale, in eighth notes. The missing letters correspond to the black keys, which aren't played in a scale.

DATA """,2

Repeat the scale twice (i.e., play the whole thing twice).

DATA """,-1

Marks the start of a new repeated passage.

DATA G,4,K,4,N,4,S,4

A few more notes, just to show quarter notes.

DATA """,4

Repeat those notes four times.

DATA @,40

A pause at the end of the piece.

All run together, it would look like this:

DATA "",-2, "scale",2

DATA G,2,I,2,K,2,L,2,N,2,P,2,R,2,S,2,"",2

DATA "",-1,G,4,K,4,N,4,S,4,"",4

DATA @,40

```

1 PG$="^P^I^A^N^O^":AU$="BY^GLEN^FISHER"
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/15/83 GF
90 GOTO 62000
100 QI=7:POKE VIC+32,11:POKE VIC+33,11:GOTO 400
104 IF TI<TM THEN 104
105 GET T$:IF T$<>"" THEN POKE VL,0:POKE SH+21,0:END
110 READ NT$,DR:IF NT$="" THEN 200
140 NT=ASC(NT$)-64:POKE SV,0:POKE EN,32:EN=EX-EN:FQ=FX-FQ
150 POKE FQ,PC(NT):POKE EN,33:TM=TI+3*DR
160 POKE SH,HP(NT):POKE SV,VP(NT):POKE SC,CL(NT):GOTO 104
200 IF DR=0 THEN RESTORE:FOR I=1 TO 24:READ X:NEXT I:GOTO 104
210 IF DR>0 THEN RP=RP+1:IF RP<DR THEN 300
220 IF DR=-2 THEN READ TL$,CL
230 RP=0:FOR I=Q0 TO Q0+5:SV(I-Q0)=PEEK(I):NEXT I
240 IF DR<-2 THEN 104
250 POKE SH+32,CL:POKE SH+33,CL
260 PRINT "{home 23°down 39°space}"
270 PRINT "{up}";TAB(20-LEN(TL$)/2);TL$
280 FOR I=1 TO 800:NEXT I
290 GOTO 104
300 FOR I=Q0 TO Q0+5:POKE I,SV(I-Q0):NEXT I:GOTO 104
400 CLR:NT=0:EN=0:FQ=0:NT$="":TM=TI:SV=0:DR=0:T$="":SC=0:SH=0:FX=0:
EX=0
410 GOSUB 61000
500 SID=54272:FOR I=0 TO 28:POKE SID+I,0:NEXT
505 T=3*16+2:POKE SID+6,T:POKE SID+13,T
506 T=3*16+7:POKE SID+5,T:POKE SID+12,T
510 FQ=SID+1:EN=SID+4:VL=SID+24
520 FX=(SID+1)+(SID+8):EX=(SID+4)+(SID+11)
600 SH=13*4096:SC=SH+39:SV=SH+1
610 POKE CRT+1016,13
620 FOR I=832 TO 895:POKE I,0:NEXT I
625 FOR I=0 TO 11:POKE 832+3*I,255:NEXT I
630 FOR I=0 TO 11:POKE 833+3*I,240:NEXT I
640 POKE SH,0:POKE SV,0
650 POKE SC,0
660 POKE SH+21,1:POKE SH+23,0:POKE SH+29,0:POKE SH+16,0:
POKE SH+27,0
700 PRINT "{clr 2°down}"
1000 IN=9:WH=3:BL=5:Q0=63
1020 GOSUB 2000:PRINT "{2°down}";:GOSUB 2000
1025 PRINT "{home down}";TAB(10);"PRESS ^rvs-on}RETURN{rvs-off"
    ^TO STOP"
1030 POKE FQ,0:POKE VL,15

```

```

1040 PRINT "{home}";
1050 X1=12:X2=2080:X3=FQ
1100 DIM PC(24),CL(24),HP(24),VP(24)
1110 FOR I=1 TO 12:READ CL(I):CL(I+12)=CL(I):NEXT I
1115 HS=2↑(1/12)
1116 P=32:FOR I=1 TO 24:PC(I)=INT(P+.5):P=P*HS:NEXT I
1120 DATA 0,1,0,1,0,0,1,0,1,0,1,0
1170 FOR I=1 TO 24:VP(I)=97-80*(I>12)-24*(CL(I)=0):NEXT I
1180 FOR I=1 TO 12:READ P:P=P+8*IN+21+CL(I):HP(I)=P
1185 HP(I+12)=P:NEXT I
1190 DATA 8,20,32,44,56,80,92,104,116,128,140,152
1200 GOTO 104
2000 PRINT TAB(IN);"{wht 21° T up}"
2010 FOR I=1 TO BL
2020 PRINT TAB(IN);"{rvs-on}~{blk}~{wht}~{blk}~{wht}~{M}~{blk}
~{wht}~{blk}~{wht}~{blk}~{wht}~{M}"
2030 NEXT I
2040 FOR I=1 TO WH
2050 PRINT TAB(IN);"{rvs-on}~{M}~{M}~{M}~{M}~{M}~{M}~{M}~{M}~{M}~{M}~{M}~{M}"
2060 NEXT I
2070 RETURN
2190 DATA "",-2,"WILTSHIRE~REEL",6
2200 DATA M,4,D,4,I,4,K,4,M,4,D,4,I,4,K,4,M,8,N,6,M,2,M,4,K,4,K,8
2205 DATA K,4,D,4,H,4,I,4, K,4,D,4,H,4,I,4,K,8,M,6,K,2,K,4,I,4,I,8,
  "",2
2210 DATA "",-1,P,2,I,2,I,2,I,8,I,2,H,4,F,4,F,8,N,2,H,2,H,2,H,8,H,2
2215 DATA F,4,D,4,D,8,P,2,I,2,I,2,I,8,I,2,H,4,F,4,F,6,F,2
2220 DATA F,4,N,4,F,4,H,4,I,8,I,8,"",2
2230 DATA @,1,@,40
2490 DATA "",-2,"DEVIL'S~DREAM",9
2500 DATA O,1,T,1,S,1,T,1,O,1,T,1,S,1,T,1,O,1,T,1,S,1,T,1,O,1
2505 DATA Q,1,O,1,M,1,L,1, M,1,Q,1,J,1,Q,1,M,1,Q,1,J,1,Q,1
2510 DATA M,1,Q,1,J,1,Q,1,T,1,S,1,Q,1,O,1, T,1,S,1,T,1,O,1
2515 DATA T,1,S,1,T,1,O,1,T,1,S,1,T,1,O,1,Q,1,O,1,M,1,L,1
2520 DATA M,1,Q,1,O,1,M,1,L,1,J,1,H,1,E,1, C,1,H,1,H,1,G,1,H,3,J,1
2525 DATA L,1,O,1,H,1,O,1,L,1,O,1,H,1,O,1, L,1,O,1,H,1,O,1
2530 DATA Q,1,O,1,M,1,L,1,M,1,Q,1,J,1,Q,1,M,1,Q,1,J,1,Q,1
2535 DATA M,1,Q,1,J,1,Q,1,T,1,S,1,Q,1,O,1, L,1,O,1,H,1,O,1
2540 DATA L,1,O,1,H,1,O,1,L,1,O,1,H,1,O,1,T,1,S,1,Q,1,O,1
2545 DATA M,1,Q,1,O,1,M,1,L,1,J,1,H,1,E,1, C,1,H,1,H,1,G,1,H,3
2546 DATA "",2
2550 DATA @,1,@,40
2590 DATA "",-2,"SICILIAN~TARANTELLA",5
2600 DATA F,1,H,2,I,3,K,1,I,2,H,3,I,1,H,2,F,4,M,2,H,4,M,2,H,4,M,2,H,
  4,M,2
2605 DATA I,2,H,2,F,2,A,2,F,2,H,2,I,3,K,1,I,2,H,3,I,1,H,1,F,4,M,2,H,
  4,M,2

```

```

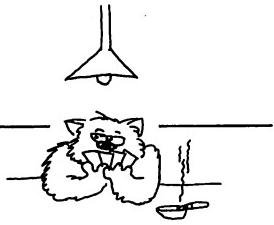
2610 DATA H,4,M,4,H,2,I,2,H,2,F,10,N,1,M,1,K,2,H,2,I,2,K,2,M,2,K,2,
I,2,F,2
2615 DATA H,2,I,2,K,2,I,2,H,2,E,2,F,2,H,2,I,2,K,2,M,2,I,2,F,2,M,4,N,
1
2620 DATA M,1,K,2,H,2,I,2,K,2,M,2,K,2,I,2,F,2,H,2,I,2,K,2,I,2,H,2,N,
2,M,2
2625 DATA K,2,I,2,H,2,F,12,S,0,R,12,S,0,R,10,M,2,M,2,N,2,M,2,K,2,I,
2,H,2,H,2
2630 DATA M,2,I,2,F,6,S,0,R,12,S,0,R,10,M,2,M,2,N,2,M,2,K,2,I,2,H,2,
F,10
2635 DATA M,2,N,2,P,2,M,2,I,2,P,2,M,2,I,2,P,2,M,2,I,2,P,2,R,2,P,2,N,
2,M,2,K,2
2640 DATA N,2,M,2,K,2,P,2,M,2,I,2,I,4,M,1,N,1,P,2,M,2,I,2,P,2,M,2,I,
2,P,2
2645 DATA M,2,I,2,P,2,O,2,P,2,T,2,R,2,P,2,N,2,M,2,K,2,I,6
2650 DATA "",2
3999 DATA @,1,@,40,"",0
60000 IN$="~":ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1); "{left}"; :ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}"; :
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}"; :
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}"
(C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120

```

```
62090 PRINT "{home 14°down}" TAB(TB)LEFT$( "{pur cyn}",K+1)LEFT$(MS$,  
    Q)  
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K  
62110 GOTO 62070  
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

BETS

By Randall Lockwood



In BETS, you play what is commonly called "Indian Poker" against three opponents (all played by the C-64). In Indian Poker, each player is dealt one card, which is held so that everyone can see everyone else's card but not their own. Everyone then places their bets, and the person with the highest card wins the round. In this game, aces have the highest value. The challenge is in betting when you don't know what card you hold.

Each player begins the game with \$200. At the start of each round, everyone throws \$10 into the pot (the *ante*). Each player matches or raises the previous bet. You can match the previous bet just by pressing RETURN, but if you choose to raise, you cannot do so by more than \$25. For example, if the outstanding bet was \$10 when your turn came, you could bet at most \$35. In addition, you may *fold*, or retire from the round, by betting \$0. By folding, you abandon any money you have in the pot, and you also abandon any chance of winning the pot, whatever card you may have.

If any player holds a card equal in value to any

opposing player and they have the highest card among all the players, then the round is considered a tie or draw (provided the player has not folded). The pot is divided evenly between those players who tied, and any remainder is left in the pot for the next round. The game ends when one player is broke. The winner is the person with the most money at the end of the game.

If you're interested in changing the ante, betting limit, or initial investment (starting cash), modify line 110 in the program. The variables in the program function as follows:

- WAD: the amount each player begins with
- ANTE: the ante of each player
- LIMIT: the betting limit for each player.

We don't expect you'll soon learn how to bet against the people the C-64 plays. While it always provides three opponents for you, it picks them out of the six it has available: Burt, Edna, Fred, Jack, Kirk, and Lucy. Each one plays and bets differently.

Note: Line 60025 is not in the standard framework.

```
1 PG$="^B^E^T^S^":AU$="BY^RANDALL^LOCKWOOD"
4 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/15/83 GF
90 GOTO 62000
100 QI=7:POKE VIC+32,0:POKE VIC+33,11
101 DEF FNR(I)=INT(RND(TI)*I)+1
110 WAD=200:ANTE=10:LIMIT=25
120 T=RND(-TI)
130 CR$="" :BD$="YOU^DON'T^HAVE^THAT^MUCH!"
240 DIM N$(7),K(3),M%(4),W%(4),B(4),S%(4),V%(4)
250 EYE=CRT+357
270 D$="{home 6°down}":D2$=D$+"{9°down}":X$=D2$+"{6°down}"
290 E$="{31°space}"
300 DIM P(4),L1(7),L2(7),L3(7)
310 FOR I=1 TO 7:READ N$(I),L1(I),L2(I),L3(I):NEXT
```

```

320 DATA JACK,14,10,9,EDNA,14,9,8,BURT,14,9,7,KIRK,13,8,5,LUCY,14,
8,4
322 DATA FRED,13,8,4,YOU,0,0,0
350 RK$="23456789{V}JQKA"
370 P(4)=7:P(1)=FNR(6)
380 P(2)=FNR(6):IF P(2)=P(1) THEN 380
400 P(3)=FNR(6):IF P(3)=P(1) OR P(3)=P(2) THEN 400
420 PRINT "{clr}":FOR I=1 TO 3:PL=P(I):PRINT D$SPC(11*I-4):::
GOSUB 1025:NEXT I
460 PRINT D2$ "~~~{rvs-on £}";:FOR I=1 TO 16:PRINT "{2°T}";:NEXT:
PRINT "*"
470 PRINT "~~{rvs-on £}";:FOR I=1 TO 17:PRINT "~~";:NEXT:
PRINT "*"
480 PRINT "yel~{rvs-on £}";:FOR I=1 TO 18:PRINT "~~";:NEXT:
PRINT "* wht"
490 PRINT "rvs-on £";:FOR I=1 TO 19:PRINT "~~";:NEXT:PRINT "*"
500 PRINT "yel rvs-on up}";:FOR I=1 TO 20:PRINT "~~";:NEXT:PRINT
510 PRINT D2$ "{down wht rvs-on}"
      SPC(7)N$(P(1))SPC(7)N$(P(2))SPC(7)N$(P(3))
520 FB=INT(RND(TI)*3+-1)
525 FOR I=1 TO 4:M%(I)=WAD:NEXT:GOSUB 1580
530 PRINT X$E$;"{5°space}":PRINT "{up wht}THE ANTE IS ~$"
      MID$(STR$(ANTE),2)
535 GOSUB 1260:PRINT X$E$
540 FOR P=1 TO 4:IF M%(P)<ANTE THEN 1860
545 NEXT P
550 FOR I=1 TO 4:B(I)=-1:Q=Q+ANTE:M%(I)=M%(I)-ANTE:NEXT:GOSUB 1580
560 S%(1)=FNR(4):V%(1)=FNR(13)
570 FOR I=2 TO 4
575 S%(I)=FNR(4):V%(I)=FNR(13)
580 FOR J=1 TO I-1:IF S%(I)=S%(J) AND V%(I)=V%(J) THEN 575
590 NEXT J:NEXT I
600 FOR I=1 TO 3:PRINT "{home}" TAB(11*I-3):::GOSUB 2000:NEXT I
730 PB=1:F=0:B=0:CL=0
740 FB=FB+1:IF FB>4 THEN FB=FB-4
750 P=FB
760 P=P+1:IF P>4 THEN P=P-4
775 OB=B(P):IF OB=0 THEN 900
780 IF P=4 THEN GOSUB 1270:GOTO 850
785 IF M%(P)<=0 THEN B(P)=0:GOSUB 2500:GOTO 850
790 IF OB<0 THEN GOSUB 2200:GOTO 850
800 GOSUB 2100
850 IF B(P)=B AND B>0 THEN CL=CL+1
860 IF B(P)>B THEN CL=1
865 IF B(P)=0 THEN F=F+1
870 IF OB<0 THEN OB=0
880 IF B(P)>0 THEN Q=Q+B(P)-OB:M%(P)=M%(P)-B(P)+OB
885 IF B<B(P) THEN B=B(P)

```

```

890 GOSUB 1580:GOSUB 1260
900 IF F+CL<4 THEN 760
970 PRINT X$E$:PB=0
980 PRINT D2$ "{6°down}YOUR CARD" TAB(24) "{3°up rvs-on 3°@ down
3°left}";:I=4:GOSUB 2000
990 GOSUB 1260:GOSUB 1260:PRINT D2$ "{6°down}" E$
1000 PRINT D2$ "{3°down}" TAB(24) "{rvs-on}^{down 3°left yel}
^{rvs-off down 3°left}^{down 3°left}^{down 3°left}
^{down 3°left 4°space}"
1010 PRINT "{home}";:FOR I=1 TO 6:PRINT E$ "^^":NEXT:PRINT X$E$
1020 GOSUB 1670:GOSUB 1260:GOTO 530
1025 ON PL GOTO 1030,1050,1070,1090,1110,1130
1030 PRINT "{rvs-on £}=={* down 5°left}={down 5°left}={Y}^{Y}
={down 6°left}<_{CHC}>{down 6°left}^{T}{down 5°left}";
1040 PRINT "{rvs-off * rvs-on}(-){rvs-off £ down 4°left rvs-on}^{*}
^{down 6°left £ 2°+ _Q£ 2°+ * down 9°left 3°+}^{Z}{3°+}:
RETURN
1050 PRINT "@@@@{down 6°left}@{@{rvs-on}^{rvs-off}@{@{down 7°left}
@{rvs-on}^{Y}^{Y}{rvs-off}@{@{down 7°left rvs-on}({C}{C}
^{down 7°left}}";
1060 PRINT "*^{V}^{*}{down 6°left rvs-off * rvs-on}^{S}{rvs-off £
down 4°left}";
1065 PRINT "{rvs-on M@N down 6°left £ 2°X 3°@ 2°X * down 9°left
9°X}":RETURN
1070 PRINT "{N 3°T M down 6°left M rvs-on 2°£}^{2°* rvs-off G
rvs-on down 6°left}^{D}{D}{down 6°left}[^{C-C}{down 6°left}
^{KEJ}{down 2°left}";
1080 PRINT "{3°left}({-}){down 6°left £MJ*KN* down 8°left £M}^{M}
^{N}{N* down 9°left}^{MNTMN}":RETURN
1090 PRINT "{N 3°T M down 6°left M- rvs-on}^{rvs-off -G down
7°left M rvs-on £F}^{R}{rvs-off G down 7°left}";
1100 PRINT "{rvs-on}({C-C}){down 6°left}^{E}{down 5°left
rvs-off * rvs-on}^{*}{rvs-off £ down 4°left}";
1105 PRINT "{rvs-on M@N down 6°left rvs-off N 2°T * rvs-on}
^{rvs-off £ 2°T M down 9°left G}^{M}":RETURN
1110 PRINT "{N 3°T M down 6°left M}^{rvs-on}^{rvs-off M down
6°left N rvs-on £U}^{U}{rvs-off M down 8°left}";
1120 PRINT "{M}{rvs-on}^{CBC}{rvs-off M down 7°left M rvs-on}^{E}
^{rvs-off N down 6°left * rvs-on J*K rvs-off £ down 4°left}";
1125 PRINT "{rvs-on M@N down 6°left rvs-on £}..{NSM}..{* down
9°left}...{GWM}...":RETURN
1130 PRINT "{rvs-on £}='={* down 5°left +}^{F}{+ down 6°left}
({U-U}){down 7°left rvs-off * rvs-on}^{CWC}{rvs-off £ down
6°left rvs-on}^{N*M}{down 2°left}";
1140 PRINT "{3°left rvs-off * rvs-on JRK rvs-off £ down 5°left
rvs-on £M@N* down 7°left £}^{MN}{MN}{* down 9°left 4°space}
.{4°space}":RETURN
1150 RETURN

```

```

1160 IF RND(1)>.04 THEN RETURN
1170 E1=E1+1:IF E1=4 THEN E1=1
1180 E=FNR(8):IF E=1 THEN EG=251:GOTO 1240
1200 IF E=2 THEN EG=254:GOTO 1240
1210 IF E=3 THEN EG=236:GOTO 1240
1220 IF E=4 THEN EG=173:GOTO 1240
1230 EG=252
1240 E=EYE+11*E1:POKE E,EG:POKE E+2,EG:RETURN
1260 FOR Z=1 TO 90:GOSUB 1160:NEXT:RETURN
1270 GET IN$:IF IN$<>"" THEN 1270
1275 PRINT X$E$:PRINT "{up 8°right}YOUR BET?~";
1280 GOSUB 60000:B(4)=VAL(IN$)
1285 IF LEFT$(IN$,1)="Q" THEN 1430
1290 IF IN$="" AND B<=0 THEN 1270
1295 IF IN$="" THEN B(4)=B:PRINT B;
1300 IF B(4)=0 THEN RETURN
1310 PRINT "{up}"
1330 IF B(4)<B THEN PRINT E$:PRINT "{up}THE BET IS ~" B:GOSUB 1260:
    GOTO 1270
1340 IF B(4)-B>LI THEN PRINT E$:PRINT "{up}RAISE LIMIT ~" LI:
    GOSUB 1260:GOTO 1270
1350 IF B(4)>M%(4) THEN PRINT E$CR$:PRINT "{up}" BD$:GOSUB 1260:
    GOTO 1270
1420 RETURN
1430 PRINT "{clr 2°down}YOU LEFT THE GAME WITH";M%(4); "DOLLARS":END
1580 PRINT X$ "{2°down wht}YOU HAVE $" MID$(STR$(M%(4)),2);
    "{6°space}"
1590 PRINT D2$;"{2°down 6°right yel rvs-on}" E$;"{up}":
    PRINT "{2°right rvs-on}HAS";
1600 FOR J=1 TO 3:PRINT TAB(11*j-4);"$";MID$(STR$(M%(J)),2);:NEXT:
    PRINT
1610 PRINT "{5°right rvs-on wht}";E$;"{up}":PRINT "{2°right rvs-on}BET";
1620 FOR J=1 TO 3:PRINT TAB(11*j-4);"{rvs-on}";
1630 IF B(J)>0 AND PB>0 THEN PRINT "$" MID$(STR$(B(J)),2);
1635 IF B(J)=0 AND PB>0 THEN PRINT ".{2*}";
1640 NEXT J
1650 PRINT:PRINT TAB(13) "{up yel rvs-on}POT:{7°space 6°left}$"
    MID$(STR$(Q),2); "{wht}":RETURN
1670 FOR I=1 TO 4:W%(I)=0:NEXT
1680 FOR I=1 TO 4:FOR J=1 TO 4:IF J=I THEN 1710
1690 IF B(I)<=0 THEN 1720
1700 IF V%(I)>=V%(J) OR B(J)=0 THEN W%(I)=W%(I)+1
1710 NEXT J
1720 NEXT I
1730 PRINT X$ "{wht}WINNER:~";:NW=0:FOR I=1 TO 4
1750 IF W%(I)<3 THEN 1790
1760 NW=NW+1:IF NW>1 THEN PRINT ".AND.";

```

```

1780 PRINT N$(P(I));
1790 NEXT:W=INT(Q/NW)
1810 PRINT "^ $" MID$(STR$(W),2):J=INT(Q/NW):PRINT E$
1820 FOR I=1 TO 4:IF W%(I)=3 THEN M%(I)=M%(I)+J:Q=Q-J
1830 NEXT:GOSUB 1260:GOSUB 1580
1840 FO=0:FOR P=1 TO 4:IF M%(P)<=0 THEN 1860
1850 NEXT:RETURN
1860 IF P=4 THEN PRINT X$;E$:PRINT X$ " {rvs-on} YOU ARE BROKE!!":
  GOSUB 1260:GOTO 1890
1870 PRINT X$;E$:PRINT X$;N$(P(P));"^ IS BROKE":GOSUB 1260:
  PRINT X$;E$
1880 GOSUB 1260
1890 I=1:FOR J=2 TO 4:IF M%(J)>M%(I) THEN I=J
1895 NEXT:PRINT "{home 2°down} THE WINNER IS ^";N$(P(I));"^ WITH ^";
  MID$(STR$(M%(I)),2);"."
1900 PRINT X$;E$:PRINT X$ " PLAY AGAIN? ^ ";:GOSUB 60000
1910 IF LEFT$(IN$,1)<>"N" THEN PRINT "{clr}":CLR:GOTO 100
1920 PRINT "{clr 4°down}":END
1930 PRINT X$ "{2°down} YOU HAVE ^" MID$(STR$(M%(4)),2);"{6°space}":
  RETURN
2000 V$=MID$(RK$,V%(I),1):IF V%(I)=9 THEN V$="10"
2010 PRINT LEFT$("{rvs-on}"+V$+"^",4) "{down 3°left} ^ {down
  3°left} ^ ";
2020 PRINT MID$("{SAZX}",S%(I),1);"^ {down 3°left} ^ {down 3°left}";
  RIGHT$(" ^ "+V$,3)
2030 RETURN
2100 IF B<=M%(P)+B(P) AND B<=3*K(P)*B(P) THEN B(P)=B:GOTO 2500
2110 B(P)=0:GOTO 2500
2200 F1=0:F2=0:F3=0:FOR I=1 TO 4:IF I=P THEN 2250
2220 IF V%(I)>=L1(P(P)) THEN F1=1
2230 IF V%(I)>L2(P(P)) THEN F2=1
2240 IF V%(I)<L3(P(P)) THEN F3=F3+1
2250 NEXT I
2260 IF F1=1 THEN B(P)=0:FO=1:GOTO 2330
2270 IF F1=0 AND FO=1 THEN K(P)=3:GOTO 2310
2280 IF F2=1 THEN K(P)=.5:GOTO 2310
2290 IF F3=3 THEN K(P)=2:GOTO 2310
2300 K(P)=1
2310 K=K(P):B(P)=INT(2*K*(5+RND(1)))
2320 IF B(P)>M%(P) THEN B(P)=M%(P)
2325 IF B(P)-B>LIMIT THEN B(P)=B+LIMIT
2330 IF B<=0 THEN 2500
2340 IF B(P)<B/3 OR B>M%(P) THEN B(P)=0:GOTO 2500
2350 IF B(P)<1.8*B THEN B(P)=B:GOTO 2500
2370 IF B(P)>B+M%(P) THEN B(P)=B+M%(P)
2500 IF B(P)>M%(P) THEN B(P)=M%(P)
2505 PRINT X$E$X$N$(P(P));
2510 IF B(P)>B AND B>0 THEN PRINT ". RAISES BET TO ";B(P):RETURN

```

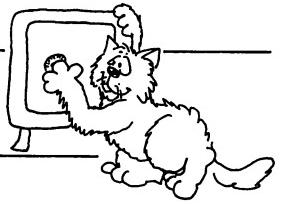
```

2520 IF B(P)=0 THEN PRINT "^FOLDS.":RETURN
2530 IF OB>0 AND B(P)>OB THEN PRINT "^SEES^RAISE.":RETURN
2540 PRINT "^BETS^$";MID$(STR$(B(P)),2):RETURN
60000 IN$="^":ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("^{\u00b1}",ZC,1);:{left}":ZC=3-ZC:
    ZT=TI+15
60025 GOSUB 1160
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "^{\left}";:
    GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
    GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
    GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
    CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
    TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:$:PRINT "{rvs-on}"
    TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
    MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:$:PRINT "{7°down cyn 9°space}
    (C)^1983^THE^CODE^WORKS"
62070 GET IN$:$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
    Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

SAFE (Uses joystick)

By Gary Marsa



Here's a way to support your computer blackjack habit: steal the money from a computer-simulated safe! In this game, you have two minutes to open a safe. Use the joystick to turn the knob. Pushing the joystick left turns the knob left, and pushing it right turns the knob right. When you think you have solved the combination, press the joystick button to open the safe.

The combination to the safe has four numbers. To find these numbers, turn the knob right to the first number, left to the second, right to the third, and left again to the fourth and final number. As you turn the

knob, you'll hear the lock click. Whenever you reach the next number in the combination, you'll hear a double click, and a chalk mark will be drawn on the safe. After you have all four numbers and four chalk marks, press the button to open the safe.

Since you have only two minutes to break the safe, you must work fast and carefully. If you overshoot a number, the lock is reset, and you must start over at the first number of the combination. (The chalk marks are erased in that case.)

Exceed the two-minute limit, and the burglar alarm goes off, and so will you—to the county jail.

```
1 PG$="^S^A^F^E^":AU$="BY^GARY^MARSA":JF=1
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/16/83 GF
90 GOTO 62000
100 DIM JD(15):POKE VIC+32,0:POKE VIC+33,0:C$="{home 6°down
18°right}"
101 FOR I=0 TO 15:READ JD(I):NEXT:DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,
8,5
102 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
16)=0
103 FOR I=0 TO 28:POKE SID+I,0:NEXT:POKE SID+6,240:POKE SID+4,17:
VL=SID+24
104 VM=CRT:CM=13.5*4096:POKE VIC+33,0:POKE VIC+27,0
110 DIM NT(11),DR(11):FOR I=1 TO 11:READ NT(I),N:DR(I)=N*7:NEXT
115 DATA 1,1, 6,1, 10,1, 6,1, 1,1, 6,1, 1,1, 3,1, 6,1, 5,1, 6,1
120 P=20:FOR I=1 TO 10:PT(I)=INT(P+.7):P=P*2↑(1/12):NEXT
125 GOSUB 1900
130 S0=SID+1:S=255:POKE VL,7:POKE S0,0
150 POKE VIC+33,0:POKE VIC+21,15:PRINT "{clr}";
155 FOR I=0 TO 17:PRINT "{rvs-on yel 39°space}" "{left inst
{left}}":NEXT
160 T$="{rvs-on}..{rvs-off}..":T$=T$+T$+T$:T$=T$+T$+T$:
L$="{rvs-on} {left inst}":R$="{rvs-on}.."
165 PRINT "{red}";:FOR I=1 TO 3:PRINT T$;R$:PRINT "{2°right}";T$;
L$:NEXT:PRINT T$;R$;"{home}"
```

```

170 PRINT "{home yel}";TAB(17) "{rvs-on D 4° I F}":
    PRINT TAB(17) "{K 4° space rvs-on K}"
171 PRINT TAB(17) "{rvs-on C rvs-off 4° I rvs-on V pur}"
175 FOR I=1 TO 17:PRINT TAB(10);"{rvs-on 20° space}":NEXT:
    PRINT "{home 3° down}"
190 PRINT TAB(11) "{rvs-on A 16° * S}"
200 PRINT TAB(11) "{rvs-on -}" TAB(28) "{rvs-off}~"
210 PRINT TAB(11) "{rvs-on -}" TAB(28) "{rvs-off}~"
220 FOR I=1 TO 3:PRINT TAB(11) "{rvs-on -}";TAB(28);{"-"}:NEXT
230 PRINT TAB(11) "{rvs-on - right UI}";TAB(28) {"-"}
240 PRINT TAB(11) "{rvs-on - right JK}";TAB(28) {"-"}
250 PRINT TAB(11) "{rvs-on -}~{D}";TAB(28) {"-"};TAB(36);{"yel
    rvs-on @ pur}""
260 PRINT TAB(11) "{rvs-on -}~{C rvs-off 2° I rvs-on V}";TAB(28)
    {"-"};TAB(35);{"yel rvs-on M rvs-off Q rvs-on G pur}"
270 PRINT TAB(11) "{rvs-on -}";TAB(28) {"-"};TAB(35);{"yel rvs-on
    M rvs-off Q rvs-on G pur}"
280 PRINT TAB(11) "{rvs-on -}";TAB(28) {"-"};TAB(36);{"yel rvs-on
    T}"
290 PRINT "{rvs-on 10° * right pur -}";TAB(28) "{rvs-off}~{right
    yel rvs-on 8° * right}";CHR$(20);{"* left inst *}"
300 PRINT TAB(11);{"pur rvs-on -"};SPC(16) "{rvs-off}~"
310 PRINT TAB(11) "{rvs-on Z 16° * X}";TAB(37);{"blu rvs-off U*
    left inst * pur}"
320 PRINT TAB(10);{"rvs-on 20° @"};TAB(34);{"yel rvs-off}={blu
    rvs-on}~{rvs-off * K}"
330 PRINT TAB(11) "{pur * rvs-on}~{rvs-off £}";TAB(26) {"* rvs-on}
    ~{rvs-off £}"
340 PRINT CS "up left rvs-on D 4° I F down 6° left rvs-off K
    4° space rvs-on K down 6° left C rvs-off 4° I rvs-on V}"
350 SA=CRT+6*WD+23:L=0:GOSUB 1000:R=-1:Y=0
355 U1=42:U2=41:U3=64
360 FOR I=1 TO 4:C(I)=INT(50*RND(1)):NEXT
370 CO$="":FOR I=1 TO 4:S$="R":IF I/2=INT(I/2) THEN S$="L"
380 CO$=CO$+S$+MID$(STR$(C(I)),2):IF I<4 THEN CO$=CO$+"-"
390 NEXT:X=INT(49*RND(1)+1)
400 FOR I=1 TO 2:FOR J=0 TO 49:PRINT CS;:IF J<10 THEN PRINT "~";
410 PRINT J "{left}~":NEXT:NEXT
420 FOR J=0 TO X:PRINT CS;:IF J<10 THEN PRINT "~";
430 PRINT J "{left}~":NEXT
440 TL=7140:TI$="000000"
450 Z=INT((7200-TI)/60+.5):T=INT(Z/60):
    QT$=MID$(STR$(1000000+T*100+Z-60*T),3)
455 PRINT "{home down}";TAB(18)MID$(QT$,4,1) ":" RIGHTS(QT$,2)
460 IF Y=1 THEN L=L+1:GOSUB 1000:R=NOT R:Y=0:GOSUB 1050
470 T=PEEK(JS) AND 12:IF T=4 THEN 900
480 IF T=8 THEN 810
485 GET E$:IF E$="Q" THEN L=0:GOSUB 1000:GOTO 1300

```

```

487 IF FNB(0) AND L=4 THEN 735
490 IF TI-T<TL THEN 450
500 L=0:GOSUB 1000
505 FOR I=1 TO 60:POKE S0,80:FOR J=1 TO 10:NEXT
510 POKE S0,0:FOR J=1 TO 10:NEXT:NEXT
520 PRINT "{home yel rvs-on U 8°* I}":PRINT "{rvs-on -}
  ^TURN ^{-}":PRINT "{rvs-on -} AROUND {-}"
530 PRINT "{rvs-on -} SLOWLY.{-}":PRINT "{rvs-on -} {6°*} {-}":
  PRINT "{rvs-on -} YOU'RE {-}"
540 PRINT "{rvs-on -} UNDER ^{-}":PRINT "{rvs-on -} ARREST! {-}":
  PRINT "{rvs-on J 3°* R 4°* K}"
550 PRINT "{rvs-on 4°* K}":FOR I=1 TO 3000:NEXT
560 FOR I=1 TO 12:FOR J=175 TO 117 STEP -1:POKE S0,J:NEXT
570 FOR J=117 TO 175:POKE S0,J:NEXT:NEXT:POKE S0,0
590 GOSUB 1600:GOSUB 1500
675 FOR J=1 TO 7000:NEXT
680 GOTO 1300
710 QT$=MID$(STR$(1000159-VAL(TI$)),3,7)
715 PRINT "{home down}";TAB(18)MID$(QT$,4,1) ":" RIGHTS(QT$,2):
  IF TI-T<TL THEN 730
720 GOTO 500
730 IF FNB(0)=0 THEN 710
735 TT=VAL(TI$):L=0:GOSUB 1000
740 PRINT "{home 12°down}";TAB(13) "{rvs-on I}":FOR I=1 TO 2:
  PRINT TAB(13) "{rvs-on K} ^{-}":NEXT
750 PRINT TAB(13) "{rvs-on V}":FOR I=1 TO 1000:NEXT:GOSUB 1100:
  GOSUB 3100
760 FOR I=1 TO 4000:NEXT:GOTO 1300
810 Y=0:IF R=-1 THEN L=0:GOSUB 1000
820 X$=RIGHT$(STR$(X),2)
830 X1$=RIGHT$(STR$(X+1),2)
840 IF X=49 THEN X1$="^0"
850 PRINT C$X$ "^.":GOSUB 990:PRINT C$RIGHT$(X$,1) "^. LEFT$(X1$,
  1):GOSUB 990
860 PRINT C$ "^. X1$":GOSUB 990:PRINT C$ ". X1$ ".":GOSUB 990
870 X=X+1:IF X=50 THEN X=0
880 IF X<>C(L+1) OR R<>0 THEN 890
885 Y=1:POKE VL,15:POKE S0,S:POKE S0,0:FOR I=1 TO 25:NEXT
890 POKE S0,S:POKE S0,0:POKE VL,7:GOTO 490
900 Y=0:IF R=0 THEN L=0:R=-1:GOSUB 1000
910 X$=RIGHT$(STR$(X),2)
920 X1$=RIGHT$(STR$(X-1),2)
930 IF X=0 THEN X1$="49"
940 PRINT C$ "^. X$":GOSUB 990:PRINT C$RIGHT$(X1$,1) "^. LEFT$(X$,
  1):GOSUB 990
950 PRINT C$X1$ "^.":GOSUB 990:PRINT C$ ". X1$ .":GOSUB 990
960 X=X-1:IF X=-1 THEN X=49
970 IF X<>C(L+1) OR R<>-1 THEN 980

```

```

975 Y=1:POKE VL,15:POKE S0,S:POKE S0,0:FOR I=1 TO 25:NEXT
980 POKE S0,S:POKE S0,0:POKE VL,7:GOTO 490
990 FOR I=1 TO 50:NEXT:RETURN
1000 IF L=0 THEN FOR Z=1 TO 7 STEP 2:POKE VIC+Z,0:NEXT:RETURN
1010 POKE VIC+L+L-1,116
1020 REM POKE VIC+38+L,1:FOR Z=1 TO 200:NEXT
1030 POKE VIC+38+L,6:RETURN
1050 IF NS=0 THEN RETURN
1055 FOR J=1 TO 110:NEXT:RETURN
1100 PRINT "{home 4°down 11°right}";:CU$="{14°up}"
1110 D$=CU$+"{I down left rvs-on}.{down left}.{down left}.{down
    left}.{down left}.{down left}.{down left}.{down left}.{down
    left}.{down left}.{down left}.{down left}.{down left}.{down
    left} Q rvs-off left}"+CU$
1120 PRINT "{A down left - down left - down left - down
    left - down left - down left - down left - down
    left - down left - down left - down left Z}";D$;
1130 PRINT "{* down left}.{down left}.{down left grn rvs-on Y down
    left Y down left Y down left rvs-off pur Y down left}.{down
    left}.{down left yel rvs-on T down left rvs-off pur Y down
    left red Q down left Q down left pur *}";D$;
1140 PRINT "{* down left}.{down left}.{down left wht rvs-on Y down
    left Y down left Y down left rvs-off pur Y down left}.{down
    left}.{down left yel rvs-on T down left rvs-off pur Y down
    left red Q down left Q down left pur *}";D$;
1150 PRINT "{* down left}.{down left grn rvs-on Y down left Y down
    left Y down left Y down left rvs-off pur Y down left}.{down
    left yel rvs-on T down left T down left rvs-off pur Y down
    left red Q down left Q down left pur *}";D$;
1160 PRINT "{* down left}.{down left wht rvs-on Y rvs-off down left}
    .{down left}.{down left}.{down left pur Y down left}.{down
    left yel rvs-on T down left T down left rvs-off pur Y down
    left}.{down left}.{down left}.{down left *}";D$;
1170 PRINT "{* down left}.{down left grn rvs-on Y down left Y down
    left Y down left Y down left rvs-off pur Y down left yel
    rvs-on T down left T down left rvs-off}.{down left pur Y down
    left blu Q down left Q down left Q down left pur *}";D$;
1180 PRINT "{* down left}.{down left}.{down left wht rvs-on Y down
    left Y down left Y down left rvs-off pur Y down left yel
    rvs-on T down left T down left T down left rvs-off pur Y down
    left blu Q down left Q down left Q down left pur *}";D$;
1190 PRINT "{* down left}.{down left grn rvs-on Y down left Y down
    left Y down left Y down left rvs-off pur Y down left yel
    rvs-on T rvs-off down left}.{down left rvs-on T down left
    rvs-off pur Y down left blu Q down left Q down left Q down
    left pur *}";D$;
1200 PRINT "{* down left}.{down left wht rvs-on Y rvs-off down left}
    .{down left}.{down left}.{down left pur Y down left yel rvs-on
    "

```

```

T down left T down left T down left rvs-off pur Y down left}
~{down left}~{down left}~{down left *}";D$;
1210 PRINT {"* down left}~{down left grn rvs-on Y down left Y down
left Y down left Y down left rvs-off pur Y down left}~{down
left yel rvs-on T down left T down left rvs-off pur Y down
left grn Q down left Q down left Q down left pur *"};D$;
1220 PRINT {"* down left}~{down left}~{down left wht rvs-on Y down
left Y down left Y down left rvs-off pur Y down left}~{down
left yel rvs-on T rvs-off down left}~{down left pur Y down
left grn Q down left Q down left Q down left pur *"};D$;
1230 PRINT {"* down left}~{down left grn rvs-on Y down left Y down
left Y down left Y down left rvs-off pur Y down left}~{down
left yel rvs-on T down left T down left rvs-off pur Y down
left grn Q down left Q down left Q down left pur *"};D$;
1240 PRINT {"* down left}~{down left wht rvs-on Y rvs-off down left}
~{down left}~{down left}~{down left pur Y down left}~{down
left}~{down left yel rvs-on T rvs-off down left pur Y down
left}~{down left}~{down left *}";D$;
1250 PRINT {"* down left}~{down left grn rvs-on Y down left Y down
left Y down left Y down left rvs-off pur Y down left}~{down
left}~{down left yel rvs-on T rvs-off down left pur Y down
left wht Q down left Q down left Q down left pur *"};D$;
1260 PRINT {"* down left}~{down left}~{down left wht rvs-on Y down
left Y down left Y down left rvs-off pur Y down left}~{down
left}~{down left yel rvs-on T rvs-off down left pur Y down
left wht Q down left Q down left Q down left pur *"};D$;
1270 PRINT {"* down left}~{down left}~{down left grn rvs-on Y down
left Y down left Y down left rvs-off pur Y down left}~{down
left}~{down left Y down left wht Q down left Q
down left Q down left pur *"};D$;
1285 RETURN
1300 GM=GM+1:POKE VIC+33,6:POKE VIC+21,0
1305 PRINT {"clr wht 2°down}THE COMBINATION WAS {rvs-on}" CO$
1310 GET T$:IF T$<>"" THEN 1310
1320 SQ$="S":IF WG=1 THEN SQ$=""
1325 IF TT>0 THEN WQ=WQ+1
1327 PRINT {"2°down}GAMES PLAYED:" GM
1330 IF WG THEN PRINT {"2°down}YOU SUCCEEDED" WQ "TIME" SQ$
1340 IF TT>0 AND (TT<BS OR BS=0) THEN BS=TT
1345 IF BS=0 THEN 1390
1360 IF TT=0 THEN 1380
1370 PRINT {"down}TIME THIS GAME:" TAB(12)INT(TT/100) "{left}:
RIGHT$(STR$(TT),2) "{down}"
1380 PRINT "YOUR BEST TIME:" TAB(12)INT(BS/100) "{left}:
RIGHT$(STR$(BS),2) "{down}"
1390 PRINT {"2°down"}:GOSUB 2000
1400 TT=0
1410 IF YN THEN 150

```

```

1420 POKE VL,0:PRINT "{clr wht}":END
1500 POKE VIC+32,11:I$="~{left inst}~"
1510 PRINT "{home cyn}";B$I$:PRINT B$I$
1520 FOR I=1 TO 18
1530 PRINT "{rvs-on 8°space blk}";
1535 FOR J=1 TO 4:PRINT "{4°right}~";:NEXT
1539 PRINT TAB(32);"{cyn 6°space}";I$
1540 NEXT I:PRINT B$;I$:PRINT "{blk}" B$;I$
1545 PRINT "{rvs-on}~~~YOUR SENTENCE:~TEN~YEARS~IN~PRISON~";I$
1550 PRINT B$;I$:
1560 PRINT "{cyn}";B$;I$; "{home wht}"
1570 RETURN
1600 B$="{rvs-on 38°space}":POKE VIC+33,14
1605 PRINT "{clr blu}"
1610 PRINT "{5°right rvs-on wht £ 13°space *_}"
1620 PRINT "{5°right rvs-on wht 15°space rvs-off 8°space yel *_"
    ~{rvs-on}~"
1630 PRINT "{5°right rvs-on wht 14°space rvs-off £ 8°space yel}~~{*"
    rvs-on}~"
1640 PRINT "{5°right rvs-on wht 8°space rvs-off £ 15°space yel N}"
1650 PRINT "{31°space yel -}"
1660 PRINT "{13°space grn rvs-on £}~{*}"
1670 PRINT "{13°space rvs-on}~"
1680 PRINT "{12°space rvs-on £}~{*}"
1690 PRINT "{12°space rvs-on 5°space}"
1700 PRINT "{12°space *_ rvs-on}~{rvs-off £}"
1710 PRINT "{13°space rvs-on}~"
1720 PRINT "{13°space *_ rvs-on}~{rvs-off £}"
1730 PRINT "{14°space wht rvs-on}~"
1740 PRINT "{14°space wht rvs-on}~"
1750 PRINT "{13°space wht rvs-on £}~{*}"
1760 PRINT "{grn}";B$
1770 PRINT B$
1780 PRINT B$
1790 PRINT B$
1800 PRINT B$
1810 PRINT B$
1899 RETURN
1900 SB=14*64
1910 FOR I=0 TO 63:POKE SB+I,0:NEXT I
1920 FOR I=0 TO 15:POKE SB+3*I,192:NEXT I
1930 SP=CRT+1016
1940 FOR I=0 TO 3:POKE SP+I,14:NEXT I
1950 FOR I=0 TO 3:POKE VIC+39+I,6:NEXT I
1960 FOR I=0 TO 3:POKE VIC+I+I,171+7*I:POKE VIC+I+I+1,0:NEXT
1970 POKE VIC+21,15
1980 RETURN
2000 NO=0

```

```

2010 TM=0:SW=1:PRINT "{wht}TRY AGAIN?{shift-space cyn}YES^NO{up}"
2020 IF TI<TM THEN 2050
2030 PRINT TAB(11+NO*4);MID$("{wht cyn}",SW,1);
MID$("YESNO{shift-space}",3*NO+1,3);"{up}"
2040 SW=3-SW:TM=TI+15
2050 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
2060 IF JB AND JD=15 THEN YN=(NO=0):RETURN
2070 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 2010
2080 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 2010
2090 GOTO 2020
3100 POKE SID+12,0:POKE SID+13,240:POKE SID+7,0:POKE SID+8,0:
POKE SID+11,17
3105 POKE SID+24,15
3110 FOR I=1 TO 11
3120 POKE SID+8,PT(NT(I)):TM=TI+DR(I)
3130 IF TI<TM THEN 3130
3140 NEXT I
3150 POKE SID+11,16:POKE SID+11,0
3160 RETURN
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("^.{+}",ZC,1);"{left}";:ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT ".{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
(C)^1983^THE^CODE^WORKS"

```

```
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```



BOSWAIN

By Chris Nadovich

TANSTAAFL—There Ain't No Such Thing As A Free Lunch. In other words, everything must be paid for, in its own way. Games of chance are no different, and the time has come to pay!

BOSWAIN is the guardian of the Temple of Yabba, and you are now his captive. BOSWAIN is willing to set you free again—for a price. You may pay in rubies (if you can win them from him), or you may pay with something else.

```
1 PG$="^B^O^S^W^A^I^N^":AU$="BY^C.T.^NADOVICH"
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/15/83 GF
90 GOTO 62000
100 QI=7:L1=0:DX=0:SC=0:LH$="*":RH$="^":POKE VIC,62
105 POKE VIC+32,0:POKE VIC+33,0
110 GOSUB 4100
170 DL=.5:GOSUB 6000:NH=1:GOSUB 11200:GOSUB 1000:DL=.5:GOSUB 6000
200 DL=.5:GOSUB 6000:HS=0:GOSUB 2000
210 IF PP=0 THEN RH$="*":LH$="^":POKE VIC,180
215 IF PP=1 THEN RH$="^":LH$="*":POKE VIC,62
220 GOSUB 4000:POKE QL,12:PRINT "{up}":PRINT TAB(29) "GUESS!":
   GOSUB 6500
230 IF Q$="Q" THEN PRINT "{clr}":GOTO 640
250 IF Q$<>"R" AND Q$<>"L" THEN GOSUB 1700:GOTO 200
270 LP=0:IF Q$="L" THEN LP=1
280 IF LP=PP THEN SC=SC-1
290 IF LP<>PP THEN SC=SC+1
300 HS=1:IF Q$="R" THEN HS=2
320 GOSUB 2000:GOSUB 7000
330 IF HS=5 THEN 340
335 HS=3:GOSUB 2000:DL=1:GOSUB 6000:HS=5:GOSUB 2000:GOSUB 4000
340 IF SC=10 GOTO 600
350 IF SC=-10 GOTO 400
360 GOSUB 11200:GOTO 200
400 RESTORE
410 READ Q$:IF Q$<>"*" THEN 410
430 GOSUB 5000:DL=2:GOSUB 6000
440 GOSUB 4900:GOSUB 13000
450 DL=5:GOSUB 6000:PRINT "{clr 2°down}":END
600 RH$="^":LH$="^":HS=3:GOSUB 2000
620 Q$="YOU ^ HAVE ^ WON!":Y=14:GOSUB 5100
```

```

630 Q$="YOU ARE FREE!":Y=12:GOSUB 5100
640 POKE VIC+21,0:PRINT "{home}":END
1000 PRINT "{clr down 6°space}OH MY!! ~ YOU SEEM TO HAVE FALLEN"
1060 PRINT "{down} ~THROUGH A TRAP DOOR. ~ ~ WHEN YOUR HEAD"
1070 PRINT "{down} ~STOPSPINNING,{shift-space}
YOUR NEW SURROUNDINGS"
1075 PRINT "{down} ~COME INTO FOCUS."
1080 PRINT "{down 6°space}ABOVE YOU IS A VAST DOMED IVORY"
1090 PRINT "{down} ~CEILING, ~TRACED WITH CRIMSON STREAKS."
1100 PRINT "{down} ~AT ITS PEAK IS A ROUND BLACK OPENING,"
1110 PRINT "{down} ~PRESUMABLY ~THROUGH ~WHICH ~YOU FELL."
1120 PRINT "{down} ~YOUR FEET TELL YOU ~THAT ~THE{shift-space}
~FLOOR,""
1130 PRINT "{down} ~HIDDEN ~BY ~{shift-space}
WAIST HIGH MIST, ~{shift-space}IS ~SOFT"
1140 PRINT "{down} ~AND ~RESILIENT."
1160 GOSUB 6400
1170 PRINT "{clr 7°down 6°space}AS ~YOU ~BEGIN ~TO ~MAKE{shift-space}
~YOUR ~WAY"
1180 PRINT "{down} ~TOWARD ~THE ~ARCHED ~WALLS, ~A ~NOISE ~FROM"
1190 PRINT "{down} ~BEHIND ~STARTLES ~YOU. ~YOU ~SPIN ~TO ~FACE"
1200 PRINT "{down} ~THE ~INHABITANT ~OF ~THIS ~EERIE ~PLACE, ~A"
1210 PRINT "{down} ~MYSTERIOUS ~HOODED ~FIGURE."
1230 GOSUB 6400
1240 PRINT "{clr}":GOSUB 9000:HS=5:GOSUB 2000
1241 RESTORE:FOR I=1 TO 3
1243 GOSUB 4900:IF I<3 THEN DL=2:GOSUB 6000
1245 NEXT I:GOSUB 6500:IF Q$="N" THEN 1600
1255 GOSUB 4900:DL=2:GOSUB 6000:GOSUB 4000:HS=3:GOSUB 2000:
GOSUB 5000
1290 DL=.5:GOSUB 6000:HS=0:GOSUB 2000:DL=1.5:GOSUB 6000:GOSUB 4900
1310 DL=2:GOSUB 6000:GOSUB 1700:Y=16:GOTO 1610
1600 GOSUB 4000:Y=16:Q$="GOOD!":GOSUB 5100:Y=Y-3
1610 Q$="LET US":GOSUB 5100:Y=Y-2:Q$="BEGIN!":GOSUB 5100:DL=3:
GOSUB 6000
1640 GOSUB 4000:RETURN
1700 PRINT "{clr 7°down 5°space rvs-on}
NOTE ~FROM ~OMNISCIENT ~NARRATOR{rvs-off}"
1710 PRINT "{2°down 6°space}YOU ~MAY ~INDICATE ~BOSWAIN'S ~HANDS"
1720 PRINT "{down} ~BY ~USING ~THE ~{rvs-on}R{rvs-off} ~AND ~{rvs-on}
L{rvs-off} ~KEYS. ~THE ~{rvs-on}R{rvs-off} ~KEY"
1730 PRINT "{down} ~INDICATES ~HIS ~RIGHT ~HAND, ~{rvs-on}L{rvs-off}
~HIS ~LEFT."
1731 PRINT "{11°space 3°T 15°space 3°T}"
1740 GOSUB 6400
1750 GOSUB 9000:HS=5:GOSUB 2000
1760 RETURN
2000 IF HS=5 THEN GOSUB 8000:CF=0:RETURN

```

```
2010 IF CF=0 THEN GOSUB 8500:CF=1
2020 ON HS GOTO 2150,2200,2250
2100 GOSUB 3200:GOSUB 3250:RETURN
2150 GOSUB 3200:GOSUB 3150:RETURN
2200 GOSUB 3100:GOSUB 3250:RETURN
2250 GOSUB 3100:GOSUB 3150:RETURN
3100 POKE QL,17:PRINT "{up}":PRINT "{2°right}";
3101 PRINT ".~.{rvs-on}~.{rvs-off}~.{down 7°left}";
3102 PRINT ".~.{rvs-on £}~.{rvs-off}~.{down 7°left}";
3103 PRINT ".~.{rvs-on £ 4°space rvs-off}~.{down 7°left}";
3104 IF LH$="*" THEN POKE VIC+1,201
3105 PRINT "{rvs-on £ rvs-off £ rvs-on H}~.{rvs-off}~.{down
    7°left}";
3106 PRINT ".~.{rvs-on 4°H rvs-off}~.{down 7°left}";
3107 PRINT ".~.{rvs-on 4°H rvs-off}~.{down 7°left}";
3108 PRINT ".~.{rvs-on 4°H rvs-off}~.{down 7°left}";
3110 RETURN
3150 POKE QL,17:PRINT "{up}":PRINT TAB(18);
3151 PRINT ".~.{rvs-on}~.{rvs-off}~.{down 7°left}";
3152 PRINT ".~.{rvs-on}~.{* rvs-off}~.{down 7°left}";
3153 PRINT ".~.{rvs-on 4°space * rvs-off}~.{down 7°left}";
3154 IF RH$="*" THEN POKE VIC+1,201
3155 PRINT ".~.{rvs-on}~.{N rvs-off * rvs-on * rvs-off down 7°left}";
3156 PRINT ".~.{rvs-on 4°N rvs-off}~.{down 7°left}";
3157 PRINT ".~.{rvs-on 4°N rvs-off}~.{down 7°left}";
3158 PRINT ".~.{rvs-on 4°N rvs-off}~.{7°left}";
3160 RETURN
3200 POKE QL,23:PRINT "{up}":PRINT "{4°right}";
3201 PRINT "{7°space 7°left up}";
3202 PRINT "{7°space 9°left up}";
3203 PRINT ".~.{rvs-on 2°N MG rvs-off}~.{up 8°left}";
3204 POKE VIC+1,0
3205 PRINT ".~.{rvs-on M}UUUU{rvs-off}~.{up 8°left}";
3206 PRINT ".~.{rvs-on M 4°@ rvs-off}~.{up 8°left}";
3207 PRINT ".~.{rvs-on £}~.{* rvs-off}~.{up 6°left}";
3208 PRINT ".~.{rvs-on}~.{rvs-off}"
3210 RETURN
3250 POKE QL,23:PRINT "{up}":PRINT TAB(18);
3251 PRINT "{7°space 7°left up}";
3252 PRINT "{7°space 7°left up}";
3253 PRINT "{rvs-on MG 2°H rvs-off 4°space up 8°left}";
3255 PRINT ".~.{rvs-on}UUUU{G rvs-off}~.{up 8°left}";
3256 PRINT ".~.{rvs-on 4°@ G rvs-off}~.{up 8°left}";
3257 PRINT ".~.{rvs-on £}~.{* rvs-off}~.{up 6°left}";
3258 PRINT ".~.{rvs-on}~.{rvs-off}"
3260 RETURN
4000 PRINT "{home}";:FOR LJ=1 TO 24:PRINT TAB(25) "{14°space}":NEXT:
    RETURN
```

```

4100 RESTORE:SB=13*64:SP=CRT+1016
4110 READ T$:IF T$<>"£" THEN 4110
4120 READ N:FOR I=0 TO N-1
4125 READ T:POKE SB+I,T:NEXT
4130 FOR I=N TO 63:POKE SB+I,0:NEXT
4140 POKE SP,13:POKE VIC+39,2
4150 POKE VIC+1,0:POKE VIC+21,1
4160 RESTORE:RETURN
4900 GOSUB 4000
5000 Y=16
5020 READ Q$:IF Q$<>". " THEN GOSUB 5100:Y=Y-2:GOTO 5020
5040 RETURN
5100 X=32-INT(LEN(Q$)/2)
5200 LN=LEN(Q$):LX=X:LY=Y
5210 FOR L=1 TO LN:POKE QL,25-Y:PRINT "{up}";
5250 PRINT TAB(LX) "*{left rvs-on}*{rvs-off left}";MIDS(Q$,L,1)
5270 LX=LX+1:NEXT L
5290 RETURN
5510 DATA "I^AM","BOSWAIN! ",.
5520 DATA "YOU^ARE^NOW","A^CAPTIVE","OF^THE^TEMPLE","YAKBAN",.
5540 DATA "NEED^I","ENLIGHTEN"
5550 DATA "YOU^ABOUT^THE","TERMS^OF^YOUR","RELEASE?","(Y^OR^N)",.
5570 DATA "TO^DEPART","YOU^MUST"
5580 DATA "PAY^A^TRIBUTE","OF^TEN^RUBIES",.
5590 DATA "IN^ONE^HAND","I^WILL^HIDE","A^GEM",.
5600 DATA "YOU^MUST","GUESS^WHICH","HAND^HOLDS","IT",.
5610 DATA =,"HA!!","YOU^HAVE","GUESSED^WRONG",.
5620 DATA "SINCE^YOU^HAVE","NO^JEWELS",.
5630 DATA "YOU^MUST^PAY","IN^FLESH!",.
5640 DATA "A^FINGER","IS^FAIR","EXCHANGE",.
5700 DATA *, "WILL^YOU","HONOR","YOUR^DEBT",..
5710 DATA "OR^MUST","I^USE","FORCE?",.
6000 X=TI+60*DL
6020 IF TI<X GOTO 6020
6030 RETURN
6400 POKE QL,24:PRINT "{up}";TAB(25);"{rvs-on}PRESS^KEY..."
6500 GET Q$:IF Q$<>"" GOTO 6500
6510 GET Q$:IF Q$="" GOTO 6510
6520 RETURN
7000 GOSUB 4000
7100 IF L1=1 OR SC<>-1 GOTO 7300
7110 L1=1:RESTORE
7120 READ Q$:IF Q$<> "=" THEN 7120
7125 FOR I=1 TO 4
7130 GOSUB 5000
7135 IF I=1 THEN HS=3:GOSUB 2000
7140 DL=2:GOSUB 6000
7145 GOSUB 4000

```

```
7150 NEXT
7155 POKE VIC+1,0
7160 PRINT "{clr 4°down 5°space rvs-on}
NOTE^FROM^OMNISCIENT^NARRATOR{rvs-off}"
7170 PRINT "{2°down 5°space}AFTER^SEVERAL^MINUTES^OF^VIGOROUS"
7175 PRINT "{down}^ARGUMENT,^BOSWAIN^AGREES^TO^ACCEPT"
7180 PRINT "{down}^YOUR^HANDS^AS^COLLATERAL^FOR^YOUR"
7185 PRINT "{down}^FINGERS.^YOU^ARE^SAFE^FOR^NOW,^BUT^IF"
7190 PRINT "{down}^YOUR^DEBT^SHOULD^BECOME^TEN^FINGERS"
7195 PRINT "{down}^THEN^OFF^COME^YOUR^HANDS."
7196 PRINT "{6°space 19°T}"
7205 GOSUB 6400
7210 GOSUB 9000:HS=5:GOSUB 2000:RETURN
7300 IF LP=PP GOTO 7308
7306 Q$="CORRECT!":Y=16:GOSUB 5100
7307 GOTO 7310
7308 Q$="INCORRECT!!":Y=16:GOSUB 5100
7310 IF SC>0 GOTO 7400
7311 IF SC=0 GOTO 7350
7315 Q$="YOU^NOW^OWE":Y=14:GOSUB 5100
7316 Q$=STR$(ABS(SC))+"_FINGER"
7320 IF SC<>-1 THEN Q$=Q$+"S"
7325 Y=12:GOSUB 5100
7330 RETURN
7350 Q$="WE^ARE^EVEN":Y=12:GOSUB 5100
7355 RETURN
7400 Q$="YOU^NOW^HAVE":Y=14:GOSUB 5100
7405 R$="^RUBIES":IF SC=1 THEN R$="^RUBY"
7410 Q$=STR$(SC)+R$:Y=12:GOSUB 5100
7420 RETURN
8000 POKE QL,12:PRINT "{up}":PRINT "{5°space Y 7°space Y 7°space T}"
8010 PRINT "{5°space H 7°space H 7°space G}^^"
8020 PRINT "{5°space - 7°space - 7°space B}^^"
8030 PRINT "{5°space B 7°space B 7°space -}^^"
8040 PRINT "{5°space G 7°space B 7°space Y}^^"
8050 PRINT "{4°space M 8°space B 7°space M}^^"
8060 PRINT "{4°space Y 17°space G}^^"
8070 PRINT "{4°space Y 17°space G}^^"
8075 POKE VIC+1,0
8080 PRINT "{4°space Y 17°space G}^^"
8085 PRINT "{4°space Y 17°space G}^^"
8086 PRINT "{4°space Y 17°space G}^^"
8087 PRINT "{4°space Y 17°space G}^^"
8090 PRINT "{home}":RETURN
8500 POKE QL,12:PRINT "{up}":PRINT "{5°space Y 7°space Y 7°space T}"
8501 PRINT "{5°space NTM 5°space H 5°space NTM}"
8502 PRINT "{4°space N shift-space}^.{shift-space M 2°shift-space}
^.{shift-space - 4°shift-space N shift-space}^.{shift-space M}"
```

```

8503 PRINT "~~~{N 5°space M 7°space N 5°space M}"
8504 PRINT "~~{Y}~{U 3°* I}~{T 5°space Y}~{U 3°* I}~{G}"
8505 PRINT "{10°space G 5°space H 8°space}"
8506 PRINT "{10°space B 5°space - 8°space}"
8507 PRINT "{10°space - 5°space B 8°space}"
8508 FOR LI=1 TO 4:PRINT "{25°space}":NEXT
8599 RETURN
9000 PRINT "{clr}"
9003 PRINT TAB(11) "{@F 2°C I}"
9004 PRINT TAB(10) "{N 4°space K}"
9005 PRINT TAB(9) "{- 6°space M}"
9006 PRINT TAB(9) "{GU CDTD*IG}"
9007 PRINT TAB(9) "{GH}~{F}~{FB-}"
9008 PRINT TAB(9) "{GY}~{V}~{VG-}"
9009 PRINT TAB(8) "{M}~{G 4°space GH}"
9010 PRINT TAB(9) "{G}~{M}~{N}~{M}"
9011 PRINT TAB(8) "{N}~{M}~{N}~{M}"
9012 PRINT TAB(7) "{N 5°space M 2°shift-space}~{M}"
9013 PRINT TAB(6) "{N 6°space M 6°space M}"
9100 RETURN
11200 IF NH<>1 GOTO 11300
11210 FOR I=0 TO 7:TH(I)=0:NEXT I
11220 CC=1:NH=0
11300 I=-2*(.5+(LP=LL))
11320 J=4*OV(2)+2*OV(1)+1*OV(0)
11330 K=TH(J)+I
11335 IF K<>0 THEN K=K/ABS(K)
11340 IF CC>4 THEN TH(J)=K
11360 OV(2)=OV(1):OV(1)=OV(0)
11370 OV(0)=1+(PP=LP)
11390 J=4*OV(2)+2*OV(1)+1*OV(0)
11400 I=TH(J)
11410 IF I=0 GOTO 11450
11420 PP=(LP-.5)*I+.5
11430 GOTO 11500
11450 I=TI
11460 PP=2*((I/2)-INT(I/2))
11500 LL=LP:CC=CC+1:RETURN
13000 POKE QL,12:PRINT "{up}"
13002 PRINT TAB(19) "{NTM}"
13003 PRINT TAB(18) "{N shift-space}~{shift-space M}"
13004 PRINT TAB(17) "{N 5°space M}"
13005 PRINT TAB(16) "{Y}~{U 3°* I}~{G}"
13006 PRINT TAB(16) "{H}~{rvs-on}~{rvs-off}"
13007 PRINT TAB(16) "{-I rvs-on £}~{*_ rvs-off}"
13008 PRINT TAB(16) "{BQ rvs-on O 4°space rvs-off I}"
13009 PRINT TAB(8) "{_* rvs-on}~{3°F 4°space L rvs-off}~{rvs-on MG
2°H rvs-off -}"

```

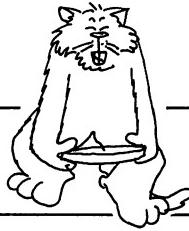
```

13010 PRINT TAB(10) "{TYU 2°Y 2°T Q* rvs-on MG 2°H rvs-off K}"
13011 PRINT TAB(17) "{J}"
13012 RETURN
53000 DATA £,41,0,255,0,3,126,64,7,131,224,15,189,240,31,126,248,60,
      255,28,114
53001 DATA 126,226,79,129,252,63,189,248,31,189,240,15,126,224,7,126,
      192,2,255
53002 DATA 0,0,168
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1); "{left}";:ZC=3-ZC:
      ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
      GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$,ZD$,Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$,:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
      GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$,:NEXT:
      GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
      CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
      TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
      TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT~RETURN~TO~BEGIN":IF JF THEN
      MS$="HIT~JOYSTICK~BUTTON~TO~BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
      (C) ~1983 ~THE~CODE~WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
      Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

HANOI

By Glen Fisher



Somewhere, in the darkest reaches of the world, the priests of an ancient cult are hard at work. At one time, their temple was in the city of Hanoi, before they fled the encroachments of modern civilization. Their goal is to transfer a large pile of handwrought silver disks from one great golden spire to another. They also have a third spire to use when they need it. It is written that when they finish moving the disks the world will end. Once you've tried the HANOI program, you'll see why it takes them so long.

In HANOI, you are presented with a pile of two to

seven brightly colored disks and three pedestals for them. The pile starts on the leftmost pedestal ("1"), and your job is to move the pile, one disk at a time, to either of the other two pedestals ("2" or "3"). There is a catch: when you move a disk, you must place it on top of a larger disk. Any attempt to place a disk on a smaller one will be thwarted by the computer. That makes the problem more difficult.

Once you've successfully moved the pile, HANOI will tell you how few moves you could have done it in, thereby justifying your pride (if you did it in that few) or humiliating you (if you didn't).

```
1 PG$="^H^A^N^O^I^":AU$="BY^GLEN^FISHER"
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/15/83 GF
90 GOTO 62000
100 QI=15:T$="{rvs-off}^{rvs-off} @ rvs-off P rvs-off O rvs-off I
      rvs-on U rvs-on Y rvs-on T}"
102 POKE VIC+32,0:POKE VIC+33,0
105 B$="{rvs-on @ rvs-on P rvs-on O rvs-on I rvs-off U rvs-off Y
      rvs-off T rvs-off}^"
110 DIM TP$(8),T$(8),BT$(8),B$(8),C$(8)
115 CR$=""
120 FOR I=1 TO 8
130 C$=MID$(T$,2*I,1):C$=C$+C$+C$:C$=C$+C$+C$:C$=LEFT$(C$+C$,14)
140 TP$(I)=MID$(T$,2*I-1,1)+C$
150 C$=MID$(B$,2*I,1):C$=C$+C$+C$:C$=C$+C$+C$:C$=LEFT$(C$+C$,14)
160 BT$(I)=MID$(B$,2*I-1,1)+C$
170 NEXT I
200 L$="{rvs-on}^{rvs-on H rvs-on K rvs-off N rvs-off}^"
210 R$="{rvs-off}^{rvs-off H rvs-off K rvs-on N rvs-on}^"
220 DIM LF$(5),RT$(5)
230 FOR I=1 TO 5:LF$(I)=MID$(L$,2*I-1,2):RT$(I)=MID$(R$,2*I-1,2):
NEXT I
500 PRINT:PRINT "{blu}HOW MANY DISKS (UP TO 7) ? ^";:GOSUB 60000:
PRINT
510 IF IN$="" THEN PRINT "LET'S USE 3":IN$="3":FOR I=1 TO 500:NEXT
```

```

520 FOR I=1 TO LEN(IN$)
530 :T$=MID$(IN$,I,1):IF T$>="0" AND T$<="9" THEN 550
540 PRINT "NO.";T$;"S.PLEASE.":GOTO 500
550 NEXT I
560 N=VAL(IN$):IF N>7 THEN PRINT "I CAN'T HANDLE MORE THAN 7.":GOTO 500
570 IF N<2 THEN PRINT "DON'T BE RIDICULOUS!":GOTO 500
600 CL$={wht red cyn pur grn blu yel}
610 FOR I=1 TO 7
620 T=INT(RND(1)*LEN(CL$)+1)
630 CL$(I)=MID$(CL$,T,1)
640 CL$=LEFT$(CL$,T-1)+MID$(CL$,T+1)
650 NEXT I
900 DN$={home 20°down}
910 RT$={14°right}
1000 FOR I=1 TO 3:FOR J=0 TO 7:P(I,J)=0:NEXT J:NEXT I
1100 PRINT "{clr}";DN$;"{up cyn}":P(1,0)=N
1110 FOR I=1 TO 5:PRINT
1115 PRINT "{rvs-on 39°space left inst}":NEXT I
1120 PRINT DN$;"{down 5°right yel rvs-on D rvs-off 2°U rvs-on F
      rvs-off down 4°left J wht}#1{yel L down 4°left rvs-on C
      rvs-off 2°O rvs-on V rvs-off 2°up}";
1130 PRINT "{9°right yel rvs-on D rvs-off 2°U rvs-on F rvs-off down
      4°left J wht}#2{yel L down 4°left rvs-on C rvs-off 2°O rvs-on
      V rvs-off 2°up}";
1140 PRINT "{9°right yel rvs-on D rvs-off 2°U rvs-on F rvs-off down
      4°left J wht}#3{yel L down 4°left rvs-on C rvs-off 2°O rvs-on
      V rvs-off up}";
1150 PRINT DN$;LEFT$("{7°up}",N);
1160 FOR I=1 TO N
1170 PRINT CL$(I);LEFT$("{rvs-off}"+RT$,7-I+1);LEFT$("{rvs-on
      @}"+BT$(1),2*I+1);"{_}"
1180 P(1,N-I+1)=I*2
1190 NEXT I
1800 MV=0
1900 TM=TI+60
1910 IF TI<TM THEN 1910
1950 GOSUB 8000
2000 PRINT "{blu}":PR$="FROM WHICH PILE?":GOSUB 5000:IF IN$="" THEN
      1950
2010 F=VAL(IN$)
2020 IF P(F,0)<1 THEN PRINT "{red} THAT PILE IS EMPTY.":GOTO 1900
2050 PR$="TO WHICH PILE?":GOSUB 5000:IF IN$="" THEN 1950
2060 T=VAL(IN$)
2070 IF F=T THEN PRINT "{red} THAT MOVE DOES NOTHING.":GOTO 1900
2080 IF P(T,0)=0 THEN 2100
2090 IF P(F,P(F,0))>P(T,P(T,0)) THEN PRINT "{red} IT WON'T FIT.":GOTO 1900

```

```

2100 FC=P(F,0):FW=P(F,FC):TC=P(T,0):TW=P(T,TC):MV=MV+1
2110 X=1+13*(F-1)+7-FW/2
2120 PRINT DN$;LEFT$(" {9°up}",FC+1);LEFT$(" {rvs-off}" +RT$+RT$+RT$,X);
2125 LF$=LEFT$(" {down 14°left}",FW+1)
2130 FOR I=1 TO 8:T$(I)=LEFT$(TP$(I),1+FW):B$(I)=LEFT$(BT$(I),1+FW):
NEXT I
2135 HT=0:J=1:IF F>T THEN J=-1
2140 FOR I=F TO T STEP J
2145 IF P(I,0)>HT THEN HT=P(I,0)
2150 NEXT I
2155 IF HT=P(T,0) THEN HT=HT+1:GOTO 2163
2160 IF ABS(F-T)>1 THEN IF HT=P(2,0) THEN HT=HT+1
2163 PRINT CL$(P(F,P(F,0))/2);
2165 FOR I=FC TO HT
2170 :FOR J=1 TO 8
2175 :PRINT T$(J);LF$;B$(J);LF$;" {2°up}";
2180 :NEXT J
2185 :PRINT "{up}";
2190 NEXT I
2195 PRINT "{down}";
2200 L$=LEFT$(" {16°left}",1+FW):R$=LEFT$(RT$,FW-1)
2201 PRINT "{rvs-on}";LEFT$(" {16°space}",FW);"{rvs-off}";
LEFT$(" {16°left}",FW);
2205 IF F>T THEN 2250
2210 FOR I=F*13 TO T*13-1
2215 :FOR J=1 TO 5
2220 PRINT LF$(J);R$;RT$(J);L$;
2225 NEXT J
2226 PRINT "{right}";
2230 NEXT I
2235 GOTO 2300
2250 FOR I=T*13 TO F*13-1
2255 PRINT "{left}";
2260 :FOR J=5 TO 1 STEP -1
2265 ::PRINT LF$(J);R$;RT$(J);L$;
2270 :NEXT J
2275 NEXT I
2300 FOR I=HT-1 TO TC STEP -1
2310 :FOR J=8 TO 1 STEP -1
2320 :PRINT T$(J);LF$;B$(J);LF$;" {2°up}";
2330 :NEXT J
2340 :PRINT "{down}";
2350 NEXT I
2410 PRINT LEFT$(" {rvs-on @}" +BT$(1),FW+1);>{@}"
2500 P(T,0)=P(T,0)+1
2510 P(T,P(T,0))=P(F,P(F,0))
2520 P(F,0)=P(F,0)-1

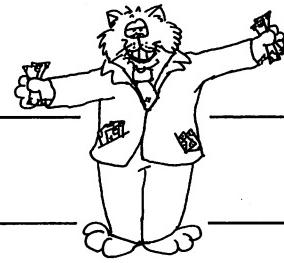
```

```

2600 IF P(2,0)<>N AND P(3,0)<>N THEN 1900
2700 GOSUB 8000
2710 PRINT "{wht down 8°space D 12°I F}"
2720 PRINT "{8°space rvs-on K}YOU DID IT!!{rvs-off K}"
2730 PRINT "{8°space C rvs-on 12°I rvs-off V}"
2740 PRINT "{down blu}IT TOOK YOU";MV;LEFT$("MOVES",5+(MV=1));
    ".TO DO IT."
2750 T=2↑N-1:PRINT "{down}THE SHORTEST POSSIBLE SOLUTION"
2760 PRINT "NEEDS";T;LEFT$("MOVES",5+(T=1));"."
2770 PRINT "{down grn}PLAY AGAIN?";:GOSUB 60000:PRINT "{clr}"
2780 IF LEFT$(IN$,1)="N" THEN END
2790 CLR:GOTO 100
2800 END
5000 PRINT PR$;
5010 GOSUB 60000:PRINT:IF IN$="" THEN RETURN
5015 IF LEFT$(IN$,1)="Q" THEN END
5016 IF LEN(IN$)>1 THEN 5030
5020 IF IN$>="1" AND IN$<="3" THEN RETURN
5030 PRINT "THERE'S NO SUCH PILE."
5040 FOR I=1 TO 500:NEXT I
5050 PRINT "{up 21°space}"
5060 PRINT "{2°up}";PR$;LEFT$("{40°space}",LEN(IN$))
5070 PRINT "{up}";:GOTO 5000
8000 PRINT "{home}";
8010 FOR I=1 TO 4:PRINT "{39°space}":NEXT I
8020 PRINT "{home}";
8030 RETURN
9000 PRINT "{home}"
60000 IN$=".":ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$(".{+}",ZC,1);"{left}";:ZC=3-ZC:
    ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT ".{left}";:
    GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
    GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
    GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
    CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
    TB=(40-LN)/2

```

```
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$ : PRINT "{rvs-on}"
    TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
    MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$ : PRINT "{7°down cyn 9°space}"
    (C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```



MISER

By Mary Jean Winter

Rumor has it that there's a lot of valuable treasure in the house where the old miser used to live. Think of it—all that treasure lying around waiting to be claimed. Surely it's worth trying to find!

After you play this adventure game, you may live to regret the idea of searching for the treasure. Or, perhaps, you may not live to regret it. Strange things await you in the old miser's house, the likes of which you've never encountered. Remember, many have gone in after the rumored "treasure," but few have come out again.

Once the program starts, give it one- or two-word commands, such as **NORTH**, or **SHOOT AARDVARK**. (As there are no guns or aardvarks in the house, we're not giving anything away.) Some other

useful commands: **INVEN** tells you what you're carrying; **SCORE** tells you your current score; **LOOK** describes the room you're in; and **QUIT** ends the game.

To move, give directions such as **GO WEST**, **WEST**, or even just **W**. All commands must have the form: verb, noun.

Keep a map of where you've been, in case you forget your way. Try synonyms for words since the author may not have thought of the word you'd use, and most important, don't assume that an object is useless just because it looks that way. You can read the program for the solution, but that's cheating. Besides, the program is so complicated that it's almost easier to solve than it is to understand.

```
1 PG$="^M^S^E^R^":AU$="BY^MARY^JEAN^WINTER"
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/15/83 GF
90 GOTO 62000
100 QI=254:DEF FNA(X)=OL%(ABS(PT%(X)))
105 POKE VIC+32,0:POKE VIC+33,0
200 DIM R$(48),R$(48,4),OM$(28),OL%(28),V$(30),O$(30),PT%(30)
210 EM=1:PF=1:FB=1:H$(1)="WHAT?":H$(2)="I^DON'T^UNDERSTAND^THAT."
300 FOR I=0 TO 48:FOR J=1 TO 4:READ R%(I,J):NEXT J:READ R$(I):
    NEXT I
400 DATA 1,0,0,0,FRONT^PORCH
401 DATA 2,0,0,12,FOYER^TO^A^LARGE^HOUSE.^DUST^IS^EVERYWHERE
402 DATA 3,1,0,0,GREAT^HALL.^SUITS^OF^ARMOR^LINE^THE^WALLS
403 DATA 0,2,4,16,BREAKFAST^ROOM.^IT^IS^BRIGHT^AND^CHEERY
404 DATA 0,5,7,3,
    CONSERVATORY.^THROUGH^A^WINDOW^YOU^SEE^A^HEDGE-MAZE
405 DATA 4,6,0,0,RED-WALLED^ROOM
406 DATA 5,0,10,0,FORMAL^PARLOR
407 DATA 0,0,8,4,GREEN^DRAWING^ROOM
408 DATA 0,9,0,7,TROPHY^ROOM.^ANIMAL^HEADS^LINE^THE^WALLS
409 DATA 8,0,0,10,DEN
```

```

410 DATA 0,11,9,6,BLUE,DRAWING,ROOM
411 DATA 10,0,0,0,LIBRARY.,EMPTY,SHELVES,LINE,WALLS
412 DATA 0,0,1,13,DINING,ROOM
413 DATA 15,0,12,0,CHINESE,ROOM
414 DATA 0,0,0,0,$
415 DATA 23,13,16,0,KITCHEN.,IT,IS,BARE
416 DATA 0,0,3,15,PANTRY.,DUST,COVERS,THE,MAHOGANY,SHELVES
417 DATA 0,8,0,18,GAME,ROOM
418 DATA 21,0,17,19,SMOKING,ROOM.,THE,AIR,IS,STALE,IN,HERE
419 DATA 21,0,18,20,
    PORTICO.,A,MURKY,POOL,GLIMMERS,ON,THE,SOUTH,SIDE
420 DATA 21,21,19,19,HALL,OF,MIRRORS,-,A,GOOD,PLACE,TO,REFLECT
421 DATA 0,19,0,20,BALLROOM.,IT,HAS,A,BEAUTIFUL,WOOD,DANCE,FLOOR
422 DATA 0,0,0,21,
    CHAPEL.,A,TABLET,SAYS,'DROP,A,RELIGIOUS,ITEM,OR,DIE!!'
423 DATA 24,15,40,25,BACK,YARD
424 DATA 24,23,24,24,FOREST
425 DATA 26,0,23,0,POOL,AREA.,THERE,IS,A,LARGE,SWIMMING,POOL,HERE
426 DATA 0,25,0,0,
    PUMP,HOUSE.,THERE,IS,POOL,MACHINERY,INSTALLED,HERE
427 DATA 35,0,31,28,MIDDLE,OF,THE,WESTERN,HALLWAY
428 DATA 0,0,27,0,WEST,BEDROOM
429 DATA 39,0,0,0,FRONT,BALCONY.,THERE,IS,A,LARGE,ROAD,BELOW
430 DATA 0,0,0,0,$
431 DATA 0,0,38,27,MASTER,BEDROOM.,THERE'S,A,HUGE,FOUR-POSTER,BED
432 DATA 0,36,0,0,REAR,BALCONY.,BELOW,YOU,SEE,A,HEDGE,MAZE
433 DATA 34,0,0,38,EAST,BEDROOM
434 DATA 0,33,0,0,CLOSET
435 DATA 0,27,36,0,
    JUNCTION,OF,THE,WEST,HALLWAY,AND,THE,NORTH-SOUTH,HALLWAY
436 DATA 32,0,37,35,CENTER,OF,THE,NORTH-SOUTH,HALLWAY
437 DATA 0,38,0,36,
    JUNCTION,OF,THE,EAST,HALLWAY,AND,THE,NORTH-SOUTH,HALLWAY
438 DATA 37,39,33,31,MIDDLE,OF,THE,EAST,HALLWAY
439 DATA 38,29,0,0,SOUTH-END,OF,THE,EAST,HALLWAY
440 DATA 0,42,0,41,HEDGE,MAZE
441 DATA 44,42,0,0,HEDGE,MAZE
442 DATA 41,44,43,0,HEDGE,MAZE
443 DATA 41,23,0,0,HEDGE,MAZE
444 DATA 0,42,0,45,HEDGE,MAZE
445 DATA 0,0,44,0,HEDGE,MAZE
446 DATA 0,0,0,5,WALK-IN,VAULT
447 DATA 0,40,0,0,DUNGEON.,THERE,IS,LIGHT,ABOVE,AND,TO,THE,SOUTH
448 DATA 0,0,0,0,
    BOTTOM,OF,THE,SWIMMING,POOL.,A,LADDER,LEADS,UP,AND,OUT
500 FOR I=1 TO 30:READ V$(I):NEXT I
510 DATA GET,TAKE,MOVE,SLID,PUSH,OPEN,READ,INVE,Q
511 DATA DROP,SAY,POUR,FILL,UNLO,LOOK
512 DATA GO,NORT,N,SOUT,S,EAST,E,WEST,W,SCOR,TURN,JUMP,SWIM,I,FIX

```

```

550 FOR I=1 TO 30:READ O$(I),PT%(I):NEXT I
560 DATA RIPE,17,MAT,10,PAPE,13,BUCK,1,SWOR,9,KEY,20,VALV,-1,LADD,
-1
570 DATA SLIP,19,RUG,15,BOOK,23,DOOR,-1,CABI,-1,RITN,-1,VICT,-1,
ORG,A,-1,PARA,14
580 DATA STAI,-1,PENN,12,CROS,11,LEAF,4,BAG,5,>$<,-1,>$<,-1,RING,7,
PAIN,8
590 DATA VAUL,-1,POOL,-1,XYZZ,-1,PLUG,-1
600 FOR I=1 TO 28:READ OM$(I),OL%(I):NEXT I:PRINT "{clr}"
610 DATA PLASTIC,BUCKET,26,VICIOUS,SNAKE,4,CHARMED,SNAKE,-2,
*GOLDEN,LEAF*,45
611 DATA *BULGING,MONEYBAG*,46,>$<,-2,*DIAMOND,RING*,48
612 DATA *RARE,PAINTING*,39,SWORD,13,MAT,0,RUSTY,CROSS,23,PENNY,28
613 DATA PIECE,OF,PAPER,31,PARACHUTE,WITH,NO,RIPCORD,34,
ORIENTAL,RUG,6
614 DATA TRAPDOOR,MARKED,'DANGER',-2
615 DATA PARACHUTE,RIPCORD,-2,PORTAL,IN,THE,NORTH,WALL,-2
616 DATA PAIR,OF,*RUBY,SLIPPERS*,2,BRASS,DOOR,KEY,-2
617 DATA MAJESTIC,STAIRCASE,LEADING,UP,2
618 DATA MAJESTIC,STAIRCASE,LEADING,DOWN,27,BATTERED,BOOK,11
619 DATA ORGAN,IN,THE,CORNER,21,OPEN,ORGAN,IN,THE,CORNER,-2
620 DATA CABINET,ON,ROLLERS,AGAINST,ONE,WALL,OVER,5,
REPAIRED,PARACHUTE,-2
621 DATA "SIGN,SAYING,'DROP,COINS,FOR,LUCK'",19
699 GOTO 14000
700 PRINT:SC=0:SF=0:GOSUB 60000:IF LEN(IN$)=0 THEN 700
720 IF LEFT$(IN$,1)="/" THEN IN$=RIGHT$(IN$,LEN(IN$)-1):GOTO 720
730 IF RIGHT$(IN$,1)="/" THEN IN$=LEFT$(IN$,LEN(IN$)-1):GOTO 730
735 SP=LEN(IN$)+1:PRINT
740 SC=SC+1:IF MID$(IN$,SC,1)="/" THEN SF=SF+1:SP=SC
750 IF SC<LEN(IN$) THEN 740
760 IF SF>-1 AND SF<2 THEN GOTO 780
770 PRINT "PLEASE,TYPE,A,ONE,OR,TWO,WORD,COMMAND":GOTO 700
780 CV$=LEFT$(IN$,SP-1):FOR X=1 TO 30:IF LEFT$(CV$,4)=V$(X) THEN
I=X:GOTO 800
790 NEXT X:GOTO 50000
800 IF SF=0 THEN CO$="":J=0:GOTO 900
810 CO$=MID$(IN$,SP+1,4):FOR X=1 TO 30:IF CO$=O$(X) THEN J=X:
GOTO 900
820 NEXT X:GOTO 50000
830 GOTO 700
900 ON I GOTO 1000,1000,2000,2000,2000,4000,5000,6000,7000,8000,
9000,10000
910 ON I-12 GOTO 11000,12000,14000,15000,16000,16000,17000,17000,
17010,17010
920 ON I-22 GOTO 19000,19000,20000,21000,22000,24000,6000,25000
1000 IF J=0 THEN 50000
1002 IF PT%(J)=-1 THEN PRINT "I,AM,UNABLE,TO,DO,THAT.":GOTO 700

```

```

1005 IF FNA(J)==-1 THEN PRINT "YOU'RE ALREADY CARRYING IT":GOTO 700
1010 IF FNA(J)<>CP THEN 51000
1020 OL%(PT%(J))=-1:PRINT "OK"
1030 X=PT%(J):IF (X>3 AND X<9) OR X=19 THEN
    PRINT "YOU GOT A TREASURE!":GT=GT+1
1040 IF J=2 AND OL%(20)==-2 THEN PRINT "YOU FIND A DOOR KEY!":
    OL%(20)=0
1045 IF J=10 AND OL%(16)==-2 THEN 2200
1050 GOTO 700
2000 IF J=0 THEN 50000
2005 IF J=13 AND CP=5 AND R%(5,3)=0 THEN 2100
2010 IF PT%(J)==-1 THEN PRINT "THAT ITEM STAYS PUT.":GOTO 700
2020 IF (FNA(J)<>CP) AND (FNA(J)<>-1) THEN 51000
2030 IF J=2 AND OL%(20)==-2 THEN 1040
2040 IF J=10 AND OL%(16)==-2 THEN 2200
2050 PRINT "MOVING IT REVEALS NOTHING."
2060 GOTO 700
2100 PRINT "BEHIND THE CABINET IS A VAULT!"
2110 FV=1:GOTO 699
2200 PRINT "YOU FIND A TRAP DOOR!"
2210 OL%(16)=6:GOTO 699
4000 IF J=0 THEN 50000
4002 IF J<>11 THEN 4030
4005 IF (FNA(J)<>CP) AND (FNA(J)<>-1) THEN 4030
4010 IN$="SCRAWLED IN BLOOD ON THE INSIDE FRONT COVER IS THE
    MESSAGE,"
4011 GOSUB 53000:PRINT
4020 PRINT "' VICTORY ' IS A PRIZE-WINNING WORD .":GOTO 700
4030 IF J=7 THEN PRINT "TRY TURNING IT.":GOTO 700
4040 IF J<>12 THEN 4120
4050 IF CP=0 AND DU=0 THEN PRINT "SORRY, THE DOOR IS LOCKED.":
    GOTO 700
4060 IF CP=0 AND DU THEN PRINT "IT'S ALREADY OPEN.":GOTO 700
4070 IF CP<>6 THEN 51000
4080 IN$="YOU OPEN THE DOOR. YOU LEAN OVER TO PEER IN, AND YOU FALL
    IN!"
4090 GOSUB 53000:CP=47:PRINT:GOTO 699
4120 IF J<>13 THEN 4160
4130 IF OL%(26)<>CP THEN 51000
4140 PRINT "THE CABINET IS EMPTY AND DUSTY."
4150 IN$="SCRIBBLED IN DUST ON ONE SHELF ARE THE WORDS, ' BEHIND
    ME '."
4155 GOSUB 53000:PRINT:GOTO 700
4160 IF J<>22 THEN 4190
4170 IF FNA(J)<>CP AND FNA(J)<>-1 THEN 51000
4180 PRINT "THE BAG IS KNOTTED SECURELY.":PRINT "IT WON'T OPEN.":
    GOTO 700
4190 IF J<>27 THEN 4230

```

```
4200 IF CP<>5 OR FV=0 THEN 51000
4210 IF VO THEN PRINT "IT'S^ALREADY^OPEN.":GOTO 700
4220 PRINT "I^CAN'T,^IT'S^LOCKED.":GOTO 700
4230 IF J<>16 THEN PRINT "I^DON'T^KNOW^HOW^TO^OPEN^THAT.":GOTO 700
4232 IF CP<>21 THEN 51000
4235 IF GG=0 THEN PRINT "IT'S^STUCK^SHUT.":GOTO 700
4240 IF OL%(24)==-2 THEN PRINT "IT'S^ALREADY^OPEN.":GOTO 700
4250 PRINT "AS^YOU^OPEN^IT,^SEVERAL^OBJECTS":
    PRINT "SUDDENLY^APPEAR!"
4260 OL%(24)==-2:OL%(25)=21:OL%(19)=21:OL%(17)=21:GOTO 699
5000 IF J=0 THEN 50000
5005 IF PT%(J)>-1 THEN IF FNA(J)<>CP AND FNA(J)<>-1 THEN 51000
5010 IF PT%(J)=-1 THEN PRINT "THERE'S^NOTHING^WRITTEN^ON^THAT.":GOTO 700
5020 IF J<>3 AND J<>11 THEN
    PRINT "THERE'S^NOTHING^WRITTEN^ON^THAT.":GOTO 700
5030 IF J=11 THEN PRINT "THE^FRONT^COVER^IS^INSCRIBED^IN^GREEK.":GOTO 700
5040 PRINT "IT^SAYS,^'12-35-6'.":
    PRINT "HMM..^LOOKS^LIKE^A^COMBINATION."
5050 KC=1:GOTO 700
6000 PRINT "YOU^ARE^CARRYING^THE^FOLLOWING:{down}"
6010 FI=0:FOR X=1 TO 27:IF OL%(X)==-1 THEN PRINT OM$(X):FI=1
6020 IF X=1 AND BF AND OL%(1)==-1 THEN
    PRINT "^THE^BUCKET^IS^FULL^OF^WATER."
6025 IF X=14 AND OL%(14)==-1 THEN PRINT "^^(BETTER^FIX^IT)"
6030 NEXT X:IF FI=0 THEN PRINT "NOTHING^AT^ALL."
6040 GOTO 700
7000 PRINT "DO^YOU^REALLY^WANT^TO^QUIT^NOW?"
7001 GOSUB 60000:IF IN$="" THEN 7001
7002 IF LEFT$(IN$,1)<>"Y" THEN PRINT:PRINT "{down}OK":GOTO 700
7005 PRINT "{clr}"
7010 PRINT "{down}YOU^ACCUMULATED";GT;"TREASURES,"
7020 PRINT "FOR^A^SCORE^OF";GT*20;"POINTS."
7030 PRINT "(100^POSSIBLE)":IF ES=0 THEN
    PRINT "{down}HOWEVER,^YOU^DID^NOT^ESCAPE."
7040 PRINT "{down}THIS^PUTS^YOU^IN^A^CLASS^OF":IF ES THEN GT=GT+1
7050 ON GT+1 GOTO 7060,7070,7075,7090,7100,7110,7115
7060 PRINT "<BEGINNER^ADVENTURER>":GOTO 7120
7070 PRINT "<AMATEUR^ADVENTURER>":GOTO 7120
7075 PRINT "<JOURNEYMAN^ADVENTURER>":GOTO 7120
7090 PRINT "<EXPERIENCED^ADVENTURER>":GOTO 7120
7100 PRINT "<PRO^ADVENTURER>":GOTO 7120
7110 PRINT "<MASTER^ADVENTURER>":GOTO 7120
7115 PRINT "<GRANDMASTER^ADVENTURER>"
7120 IF GT<>6 THEN PRINT "{down}BETTER^LUCK^NEXT^TIME!"
7150 END
8000 IF FNA(J)<>-1 THEN PRINT "YOU^AREN'T^CARRYING^IT!":GOTO 700
```

```

8010 X=PT%(J)
8015 IF (X>3 AND X<9) OR X=19 THEN PRINT "DON'T DROP *TREASURES*!":
GOTO 700
8020 IF CP=19 AND J=19 THEN 8100
8030 IF CP=22 AND J=20 THEN 8200
8040 OL%(PT%(J))=CP:PRINT "OK":GOTO 700
8100 PRINT "AS THE PENNY SINKS BELOW THE SURFACE OF"
8105 PRINT "THE POOL, A FLEETING IMAGE OF A CHAPEL"
8110 PRINT "WITH DANCERS OUTSIDE APPEARS."
8130 R%(21,3)=22:OL%(12)=-2:GOTO 700
8200 PRINT "EVEN BEFORE IT HITS THE GROUND, THE":
PRINT "CROSS FADES AWAY"
8210 PRINT "{down}THE TABLET HAS DISINTEGRATED."
8215 PRINT "{down}YOU HEAR MUSIC FROM THE ORGAN."
8220 GG=1:OL%(11)=-2:R$(22)="CHAPEL"
8221 OM$(24)="CLOSED ORGAN PLAYING MUSIC IN THE CORNER":GOTO 700
9000 IF J=0 THEN PRINT "SAY WHAT???:GOTO 700
9010 IF J=14 THEN 9100
9020 IF J=15 THEN 9200
9030 IF J>28 THEN 9300
9040 PRINT "OKAY, '^';RIGHT$(IN$,LEN(IN$)-SP);'.'"
9050 FOR X=1 TO 1000:NEXT X:PRINT "NOTHING HAPPENS.":GOTO 700
9100 IF CP<>4 OR CH THEN PRINT "NOTHING HAPPENS.":GOTO 700
9110 IN$="THE SNAKE IS CHARMED BY THE VERY UTTERANCE OF YOUR WORDS."
9111 GOSUB 53000:PRINT
9120 CH=1:OL%(2)=-2:OL%(3)=4:GOTO 700
9200 IF CP<>8 OR PO THEN PRINT "NOTHING HAPPENS.":GOTO 700
9210 PRINT "A PORTAL HAS OPENED IN THE NORTH WALL!!"
9220 PO=1:R%(8,1)=17:OL%(18)=8:GOTO 700
9300 PRINT "A HOLLOW VOICE SAYS, 'WRONG ADVENTURE'.":GOTO 700
10000 IF J<>4 THEN PRINT "I WOULDN'T KNOW HOW.":GOTO 700
10010 IF OL%(1)<>-1 AND OL%(1)<>CP THEN 51000
10020 IF BF=0 THEN PRINT "THE BUCKET IS ALREADY EMPTY.":GOTO 700
10030 IF CP=19 THEN PRINT "OK":GOTO 700
10040 IF CP=10 AND FB THEN 10050
10045 PRINT "THE WATER DISAPPEARS QUICKLY.":BF=0:GOTO 700
10050 PRINT "CONGRATULATIONS! YOU HAVE VANQUISHED":
PRINT "THE FLAMES."
10060 FB=0:BF=0:GOTO 699
11000 IF J=0 THEN 50000
11010 IF PT%(J)=-1 THEN PRINT "THAT WOULDN'T HOLD ANYTHING.":GOTO 700
11020 IF FNA(J)<>CP AND FNA(J)<>-1 THEN 51000
11030 IF J<>4 THEN PRINT "THAT WOULDN'T HOLD ANYTHING.":GOTO 700
11040 IF BF THEN PRINT "IT'S ALREADY FULL.":GOTO 700
11050 IF CP=25 AND PF THEN
PRINT "I'D RATHER LEAVE THE MERCURY ALONE.":GOTO 700
11060 IF CP<>23 AND CP<>19 THEN PRINT "I DON'T SEE ANY WATER HERE.":GOTO 700

```

```

11070 PRINT "YOUR BUCKET IS NOW FULL.":BF=1:GOTO 700
12000 IF J=0 THEN 50000
12010 IF J<>12 AND J<>27 THEN
    PRINT "I DON'T KNOW HOW TO UNLOCK ONE.":GOTO 700
12020 IF CP<>0 AND CP<>5 AND CP<>6 THEN 51000
12030 IF CP=0 AND J=12 THEN 12200
12040 IF CP=5 AND J=27 THEN 12300
12050 IF CP<>6 OR J<>12 OR OL%(16)==-2 THEN 51000
12100 PRINT "THE TRAPDOOR HAS NO LOCK":GOTO 700
12200 IF DU THEN PRINT "IT'S ALREADY UNLOCKED.":GOTO 700
12210 IF OL%(20)<-1 THEN PRINT "I NEED A KEY.":GOTO 700
12220 PRINT "THE DOOR EASILY UNLOCKS AND SWINGS OPEN.":DU=1:GOTO 699
12300 IF VO THEN PRINT "IT'S ALREADY OPEN.":GOTO 700
12305 IF FV=0 THEN 51000
12310 IF KC=0 THEN PRINT "I DON'T KNOW THE COMBINATION.":GOTO 700
12320 PRINT "OK, LET'S SEE. 12..35..6. . .":
    PRINT "<CLICK!> THE DOOR SWINGS OPEN."
12330 VO=1:R%(5,3)=46:GOTO 699
14000 IN$="{down}YOU ARE IN THE "+R$(CP)+".":GOSUB 53000:PRINT
14010 FOR X=1 TO 28:IF OL%(X)<>CP THEN 14020
14011 IN$="THERE IS A "+OMS(X)+" HERE.":PRINT:GOSUB 53000:PRINT
14020 IF X=1 AND BF AND OL%(1)=CP THEN
    PRINT " ~~ THE BUCKET IS FULL OF WATER."
14030 NEXT X
14040 IF CP=25 AND PF THEN
    PRINT "{down}THE POOL IS FULL OF LIQUID MERCURY!"
14050 IF CP<>25 OR PF THEN 14060
14055 PRINT "{down}THE POOL'S EMPTY.":IF OL%(7)<>48 THEN 14060
14056 PRINT "{down}I SEE SOMETHING SHINY IN THE POOL!"
14060 IF CP<>10 OR FB=0 THEN 14090
14070 PRINT "{down}THERE IS A HOT FIRE ON THE SOUTH WALL!"
14080 PRINT "IF I GO THAT WAY I'LL BURN TO DEATH!"
14090 IF CP<>16 THEN 14110
14100 IN$="{down}
A RICH, FULL VOICE SAYS, 'RITNEW IS A CHARMING WORD'."
14105 GOSUB 53000:PRINT
14110 IF CP=26 THEN
    PRINT "{down}THERE IS A VALVE ON ONE OF THE PIPES."
14115 IF CP=23 THEN PRINT "{down}THERE IS A LEAKY FAUCET NEARBY."
14120 IF CP=10 AND FB=0 THEN
    PRINT "{down}THERE IS EVIDENCE OF A RECENT FIRE HERE."
14125 IF CP=5 AND FV THEN
    PRINT "{down}THERE IS A VAULT IN THE EAST WALL."
14126 IF CP=5 AND VO THEN PRINT "THE VAULT IS OPEN."
14127 IF CP=0 AND DU THEN PRINT "{down}AN OPEN DOOR LEADS NORTH."
14130 IF CP<>48 THEN PRINT "{down}OBVIOUS EXITS":IF R%(CP,1)>0 THEN
    PRINT "N ";
14140 IF R%(CP,2)>0 THEN PRINT "S ";

```

```

14150 IF R%(CP,3)>0 THEN PRINT "E~";
14160 IF R%(CP,4)>0 THEN PRINT "W~";
14170 PRINT:GOTO 700
15000 IF J<>8 AND J<>18 AND J<>28 THEN 50000
15010 IF (J=8 AND CP<>48) OR (J=28 AND CP<>25) THEN 51000
15015 IF J=18 AND CP<>2 AND CP<>27 THEN 51000
15020 IF J=8 THEN CP=25:GOTO 699
15030 IF J=28 AND PF THEN PRINT "THE~POOL~IS~FULL~OF~MERCURY!":
    GOTO 700
15040 IF J=28 THEN CP=48:GOTO 699
15050 IF CP=27 THEN CP=2:GOTO 699
15060 IF OL%(9)==-1 THEN 15070
15061 IN$="THE~SUITS~OF~ARMOR~PREVENT~YOU~FROM~GOING~UP!":GOSUB 53000
15062 PRINT:GOTO 700
15070 PRINT "THE~SUITS~OF~ARMOR~TRY~TO~STOP~YOU~"
15080 PRINT "BUT~YOU~FIGHT~THEM~OFF~WITH~YOUR~SWORD.":CP=27:GOTO 699
16000 IF CP=0 AND DU=0 THEN PRINT "THE~DOOR~IS~LOCKED~SHUT.":GOTO 700
16010 IF R%(CP,1)=0 THEN 52000
16015 IF CP=0 THEN PRINT "{down}THE~DOOR~SLAMS~SHUT~BEHIND~YOU!"
16020 CP=R%(CP,1):GOTO 699
17000 IF CP=10 AND FB THEN PRINT "YOU~HAVE~BURNT~TO~A~CRISP!":END
17010 IF CP=4 AND CH=0 THEN 17060
17030 D=2:IF I=21 OR I=22 THEN D=3
17040 IF R%(CP,D)=0 THEN 52000
17050 CP=R%(CP,D):GOTO 699
17060 IF PS=0 THEN PRINT "THE~SNAKE~IS~ABOUT~TO~ATTACK!":PS=1:
    GOTO 700
17070 PRINT "THE~SNAKE~BITES~YOU!":PRINT "YOU~ARE~DEAD.":END
19000 IF R%(CP,4)=0 THEN 52000
19010 CP=R%(CP,4):GOTO 699
20000 PRINT:PRINT "IF~YOU~WERE~TO~QUIT~NOW~":
    PRINT "YOU~WOULD~HAVE~A~SCORE~OF~";
20010 PRINT GT*20;"POINTS.":PRINT "(100~POSSIBLE){down}"
20020 PRINT "DO~YOU~INDEED~WISH~TO~QUIT~NOW?~";
20030 GOSUB 60000
20040 PRINT:IF LEFT$(IN$,1)="Y" THEN 7010
20050 IF LEFT$(IN$,1)<>"N" THEN PRINT "PLEASE~ANSWER~YES~OR~NO":
    GOTO 20020
20060 PRINT "OK":PRINT:GOTO 700
21000 IF J<>7 THEN PRINT "I~DON'T~KNOW~HOW~TO~TURN~SUCH~A~THING.~":
    GOTO 700
21010 IF CP<>26 THEN 51000
21020 IN$="WITH~MUCH~EFFORT,~YOU~TURN~THE~VALVE~5~TIMES.~YOU~HEAR
    ~THE~SOUND~"
21030 IN$=IN$+"OF~LIQUID~":GOSUB 53000:PRINT:
    PRINT "FLOWING~THROUGH~PIPES."
21040 PF=1-PF
21050 IF PF=0 AND OL%(7)==-3 THEN OL%(7)=25:GOTO 700

```

```

21060 IF PF AND OL%(7)=25 THEN OL%(7)=-3:GOTO 700
21070 GOTO 700
22000 IF CP<>27 AND CP<>29 AND CP<>32 THEN 22050
22010 PRINT "YOU JUMP..":IF CP=27 THEN 22500
22020 IF OL%(14)=-1 THEN 22100
22030 IF OL%(27)=-1 THEN 22200
22040 PRINT "YOU HIT THE GROUND.":GOTO 22540
22050 PRINT "THERE'S NOWHERE TO JUMP.":GOTO 700
22100 PRINT "THERE IS NO WAY TO OPEN THE PARACHUTE!":GOTO 22040
22200 PRINT "YOU YANK THE RIPCORD AND THE"
22210 PRINT "'CHUTE COMES BILLOWING OUT.":IF CP=32 THEN CP=40:
      GOTO 699
22220 PRINT "YOU LAND SAFELY":
      PRINT "{down}CONGRATULATIONS ON ESCAPING!"
22230 ES=1:GOTO 7010
22500 IF JM THEN 22530
22510 PRINT "YOU HAVE LANDED DOWN-STAIRS.:
      PRINT "AND NARROWLY ESCAPED SERIOUS"
22520 PRINT "INJURY. PLEASE DON'T TRY IT AGAIN.":JM=1:CP=2:GOTO 699
22530 PRINT "NOW YOU'VE DONE IT! YOU IGNORED"
22535 PRINT "MY WARNING, AND AS A RESULT"
22540 PRINT "YOU HAVE BROKEN YOUR NECK!":PRINT "{down}YOU ARE DEAD. ":
      END
24000 IF CP=19 THEN PRINT "THE WATER IS ONLY A FEW INCHES DEEP. ":
      GOTO 700
24010 IF CP<>25 THEN PRINT "THERE'S NOTHING HERE TO SWIM IN!":
      GOTO 700
24020 IF PF THEN PRINT "IN MERCURY? NO WAY!":GOTO 700
24030 PRINT "THE POOL IS EMPTY.":GOTO 700
25000 IF J=0 THEN 50000
25010 IF J=7 THEN PRINT "I AIN'T NO PLUMBER.":GOTO 700
25020 IF J<>17 THEN PRINT "I WOULDN'T KNOW HOW.":GOTO 700
25030 IF FNA(J)<>CP AND FNA(J)<>-1 THEN 51000
25040 IF OL%(14)=-2 THEN PRINT "IT'S ALREADY FIXED.":GOTO 700
25050 IF OL%(17)<>-1 THEN PRINT "I NEED A RIPCORD.":GOTO 700
25060 PRINT "I'M NO EXPERT, BUT I THINK IT'LL WORK."
25070 OL%(27)=OL%(14):OL%(14)=-2:PT%(17)=27:OL%(17)=0:GOTO 700
50000 PRINT HS(EM):EM=3-EM:GOTO 700
51000 PRINT "I DON'T SEE IT HERE.":GOTO 700
52000 PRINT "IT'S IMPOSSIBLE TO GO THAT WAY.":GOTO 700
53000 IF LEN(IN$)<=40 THEN PRINT IN$,:RETURN
53005 OE=40
53010 IF MID$(IN$,OE,1)<>" " THEN OE=OE-1:GOTO 53010
53020 PRINT LEFT$(IN$,OE):PRINT RIGHT$(IN$,LEN(IN$)-OE);:RETURN
60000 IN$=" ":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$(" "+,ZC,1);"{left}";:ZC=3-ZC:
      ZT=TI+15

```

```

60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT ".{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
(C) ^1983 ^THE ^CODE ^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

MAD



By Glen Fisher

It's story time for the C-64, only this time the C-64 is telling the stories, but with your help. At various points in the story you'll be asked for a word to fill in a blank. The only hitch is that you must supply the word before you know how it will be used.

MAD will choose a story and then ask for words, or more accurately, parts of speech, such as nouns, verbs, adjectives, and so forth. Sometimes it will ask for something specific, such as a number or a disaster.

For those of you who are a trifle hazy on grammar, here's a quick summary. A noun names a thing, such as "arcade," "painting," or "hole." A verb expresses action, such as "run," "grow," or "scribble." An adjective describes a noun, such as "blue," "portable," or "grandiose." An adverb modifies verbs, adjectives, or adverbs, and usually ends in "ly." Examples are "quickly," "haltingly," and "oddly." Exclamations are sharp or sudden cries, such as "Oh!," "Darn!," or "Forsooth!" (There are a couple more, but the censor wouldn't let us use them.)

We suggest that you pick strange words that are unrelated to each other. This seems to produce the best stories.

You can replace the stories we have provided by eliminating lines 10000-11540. Add your own stories

by putting in your own DATA statements somewhere after line 10000 and before our standard framework, which starts at line 60000. If you want to add stories of your own to MAD's list, you'll need to know how it handles the "fill-in-the-blanks." Each "fill-in" is marked by a percent sign (%), followed by a one-letter code:

- %A Adverb
- %B Part of your body
- %C Disaster (calamity)
- %D Disease
- %E Exclamation
- %J Adjective
- %N Noun
- %P Plural noun
- %U Number
- %V Verb
- %Y Animal
- %Z A person's name

In addition, an "equal" sign within the text forces MAD to start a new line; two in a row will produce a blank line. The end of each story is marked by a SHIFT-Q.

```
1 PG$="^M^A^D":AU$="BY^GLEN^FISHER"
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/16/83 GF
90 GOTO 62000
100 DIM P$(26)
102 POKE VIC+32,9:POKE VIC+33,9
105 DIM M$(100)
110 FOR I=0 TO 25:READ P$(I):NEXT I
150 NS=5:DIM USD(NS)
160 FOR I=1 TO NS:USD(I)=I:NEXT I
200 SC=NS:REM RESTART
300 PRINT "{clr 3°down}"
310 IF SC=0 THEN 160
```

```

320 RESTORE:FOR I=1 TO 26:READ T$:NEXT I
330 PRINT "PLEASE ^TYPE ^IN ^THE ^FOLLOWING :{down}"
340 IF SC>(NS-3) THEN 400
350 FOR I=1 TO NS:SN=I
355 IF USD(SN)>0 THEN 365
360 NEXT I
365 USD(SN)==-1:GOTO 406
400 I=INT(RND(1)*SC)+1
401 SN=USD(I)
402 IF USD(I)<1 THEN 400
403 USD(I)==-1
406 SC=SC-1
410 I=0
500 READ S$:IF LEFT$(S$,1)<>"{Q}" THEN 500
510 I=I+1:IF I<>SN THEN 500
520 N=0
530 S$=MID$(S$,2,255)
1000 IF S$="" THEN M$(N)=M$:N=N+1:M$="":READ S$:
    IF LEFT$(S$,1)="{Q}" THEN 2000
1010 C$=LEFT$(S$,1):S$=MID$(S$,2,255)
1020 IF C$<>"%" THEN M$=M$+C$:GOTO 1000
1025 IF S$="" THEN 1000
1030 C$=LEFT$(S$,1):S$=MID$(S$,2,255)
1040 T=ASC(C$)-ASC("A")
1050 PRINT:PRINT "A";
1060 FOR I=1 TO 5
1070 IF LEFT$(P$(T),1)=MID$("AEIOU",I,1) THEN PRINT "N";
1080 NEXT I
1100 PRINT "^";P$(T);":^";
1110 GOSUB 60000:T$=IN$:IF IN$="" THEN PRINT "{2°up}":GOTO 1050
1115 IF IN$="Q" THEN PRINT "{clr down}THANKS ^FOR ^PLAYING":END
1120 M$=M$+T$:GOTO 1000
2000 I=-1:PRINT "{clr}";
2003 N=N-1:M$(N)=M$(N)+"^"
2005 I=I+1:IF I>N THEN 3000
2007 M$=M$(I)
2010 IF M$="" THEN 2005
2020 C$=LEFT$(M$,1):M$=MID$(M$,2,255)
2030 IF C$<>"^" AND C$<> "=" THEN W$=W$+C$:GOTO 2010
2050 IF LEN(L$)+LEN(W$)+1>39 THEN PRINT L$:L$=""
2060 L$=L$+"^"+W$:
2065 IF C$="-" THEN PRINT L$:L$=""
2070 W$="":GOTO 2010
3000 PRINT L$:L$=""
3010 PRINT "{down} ^WANT ^TO ^TRY ^ANOTHER? ^":GOSUB 60000:RJ$=IN$
3020 IF LEFT$(RJ$,1)="N" THEN 3500
3025 GOTO 300
3500 PRINT "{clr 4°down}THANKS ^FOR ^PLAYING .":END

```

10000 DATA ADVERB,"PART^OF^YOUR^BODY"
 10004 REM C=CALAMITY
 10005 DATA DISASTER,DISEASE,EXCLAMATION
 10010 DATA F,G,H,I,ADJECTIVE
 10020 DATA K,L,M,NOUN,O
 10030 DATA "PLURAL^NOUN",Q,R,S,T
 10040 DATA NUMBER,VERB,W,X,ANIMAL
 10050 DATA "NAME^OF^A^PERSON^YOU^KNOW"
 10065 DATA "{Q}DEAR^SIRS:=="
 10070 DATA
 "I^RECENTLY^PURCHASED^A^%J^COMPUTER^FROM^A^%J^STORE.^I^GOT^IT
 ^FOR^"
 10075 DATA "%Z,^WHO^JUST^LOVES^ALL^SORTS^OF^%J^GADGETS.^"
 10080 DATA "^WHEN^I^BROUGHT^THE^%J^COMPUTER^TO^MY^%N,^IT^WAS^SO^%J"
 10090 DATA "^THAT^IT^MADE^MY^%B^ACHE.^I^READ^YOUR^%J^MANUAL,^"
 10100 DATA "BUT^IT^DIDN'T^HELP^ME^UNDERSTAND^THE^%J^COMPUTER.^"
 10110 DATA "I^THINK^THIS^COULD^LEAD^TO^A^SERIOUS^CASE^OF^%D."
 10120 DATA
 "^EVERY^%J^PROGRAM^I^WRITE^SEEMS^TO^BE^%J.^CAN^I^EXCHANGE^"
 10130 DATA "THIS^%J^COMPUTER^FOR^ONE^THAT^WILL^LET^ME^%V"
 10140 DATA "^PROGRAMS,^AND^THAT^WON'^T^MAKE^MY^%B^SORE?"
 10150 DATA
 "==P.S.^I^HAVE^%U^OTHER^%P,^AND^THEY^ALL^WORK^BETTER^THAN^"
 10160 DATA "YOUR^%J^LITTLE^%Y!"
 11210 DATA "{Q}
 ONCE^UPON^A^TIME,^A^%J^WOMAN^FELL^IN^LOVE^WITH^HER^%J
 ^COMPUTER.^"
 11220 DATA
 "SHE^WOULD^BEGIN^EACH^DAY^BY^KISSING^ITS^%N^AND^%A^CARESSING
 ^ITS^"
 11230 DATA
 "%P.^THEN^ONE^DAY,^TRAGEDY^STRUCK:^SHE^AWOKE,^AND^FOUND^HER"
 11240 DATA
 "^COMPUTER^DESTROYED^BY^A^%C.^%E!^SHE^SAID,^HOW^CAN^I^EVER^BE^"
 11250 DATA "%J^AGAIN?^BUT^THEN^A^"
 11260 DATA
 "WONDERFUL^THING^CAME^TO^PASS:^%A,^A^BRAND^NEW^%J^COMPUTER
 ^APPEARED"
 11270 DATA "ON^HER^DESK.^IT^HAD^A^%J^USERS'S^%N,^AND^IT^WORKED^"
 11280 DATA "%A!^%E!^THE^%N^LIKES^ME!",^SHE^EXCLAIMED.=="
 11310 DATA "{Q}=HOW^TO^GET^AHEAD...=="
 11320 DATA "FIRST,^YOU^MUST^ALWAYS^TRY^TO^%V^YOUR^BOSS.^"
 11330 DATA "GO^TO^WORK^%A,^AND^ALWAYS^EAT^YOUR^%N^AT^YOUR^%N.^"
 11340 DATA "^BE^%J^WHEN^YOU^ANSWER^THE^%N,^AND^WHEN^YOU^ARE^"
 11350 DATA "ASKED^TO^%V^A^%J^JOB,^BE^SURE^AND^GIVE^IT^YOUR^"
 11360 DATA "BEST^%N.==REMEMBER,^THE^CUSTOMER^IS^ALWAYS^%J!"
 11370 DATA "{Q}==RECENT^EMPLOYEE^INJURIES:=="

```

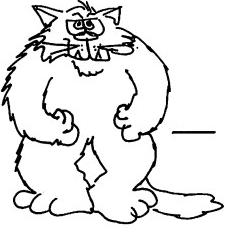
11380 DATA "AN ELECTRICAL MAINTENANCE MAN TOUCHED A %N TO A METAL %N ."
11390 DATA "THE EXPLOSION HURT HIS %B =="
11400 DATA "A PROFESSOR WAS HIT BY A %N AND SUFFERED ."
11410 DATA "BRUISES AND CUTS ON HER %B =="
11420 DATA "A FOOD SERVICE WORKER FELL WHILE MOPPING THE %N , ."
11430 DATA "INJURING HER LEGS . =="
11440 DATA "A COMPUTER PROGRAMMER FELL ASLEEP AT HER TERMINAL . ."
11450 DATA "A %C OCCURRED WHILE SHE SLEPT , "
11460 DATA " , CAUSING HER %J COMPUTER TO FALL ON HER %B =="
11470 DATA "{Q}CLASSIFIED ADS =="
11480 DATA "FREE %N LESSONS IN EXCHANGE FOR KEEPING %N ."
11490 DATA "FREE OF %N . (766-2337) =="
11500 DATA "VOLUNTEERS NEEDED FOR RESEARCH PROJECT . MUST NOT ."
11510 DATA "BE AFRAID OF %P , %P , OR A %J %N . FIND OUT MORE ."
11520 DATA "BY CALLING %Z AT 968-9905 =="
11530 DATA "SICKLY %Y FOR SALE . (976-1880) =="
11540 DATA "{Q}"
60000 IN$=" ":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<TI THEN PRINT MID$("~{+}",ZC,1);:{left};:ZC=3-ZC:
  ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
  GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$,ZD$,Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
  GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
  GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
  CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
  TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
  TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT RETURN TO BEGIN":IF JF THEN
  MS$="HIT JOYSTICK BUTTON TO BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
  (C) ^1983 ^THE ^CODE ^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120

```

```
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

GODZILLA (Uses joystick)

By Glen Fisher



Sound the alarm! Alert the armed forces! Godzilla is on the loose again! And it's up to you to stop him! Succeed, and all of Japan will be grateful. Fail, and—well, best not to think about that.

You have at your disposal the Japanese Army, Navy, and Air Force. In addition, you have ten high-explosive missiles and, as a last resort, an atomic bomb.

The left side of the screen will show a map of Japan (with Tokyo highlighted) and of the surrounding ocean. The entire map is divided into 25 sectors. Godzilla will always occupy one of the sectors. Your attacks may be launched from any sector; the sector you're attacking from will be marked with a large blue cursor. You can move the blue cursor around the map with the joystick. The weapons available in each sector, as well as the sector's population, will be printed in blue to the right of the map.

You alternate turns with Godzilla. During his turn, Godzilla can move to any of the eight sectors adjacent to his current position. In general, he will not damage troops, weapons, or the population unless provoked by an attack. However, if he moves into Tokyo, he cannot avoid doing damage because Tokyo is so crowded. During your turn, you can direct an attack or move your troops and ships between sectors by typing in the number of one of the eight commands below.

1. **Land Attack:** The troops in Godzilla's sector open fire on him. You may choose how many of the available troops to use.
2. **Sea Attack:** All the ships in Godzilla's sector fire on him.
3. **Air Attack:** Airplanes from the sector you have selected launch a raid on Godzilla. Use the joystick to move the blue cursor to the sector you want to attack from, type in the number 3, and then choose how many planes you want to take part in the attack. Planes can attack Godzilla from a distance of two sectors.

4. **Move Troops:** Moves troops from one sector to any of the eight adjacent sectors. Using the joystick, move the blue cursor to the sector you want the troops moved from, and type in the number 4. A red cursor will appear; move it to the location where you want to transfer the troops (they must stay on land), and press the joystick button. If you want to cancel the movement, press any key after the red cursor appears.
5. **Move Ship:** Moves one ship from one sector to any of the eight adjacent sectors. Using the joystick, move the blue cursor to the sector you want the ship moved from, and type in the number 5. A red cursor will appear; move it to the location where you want to transfer the ship (it must stay on the ocean), and press the joystick button. If you want to cancel the movement, press any key after the red cursor appears.
6. **Missile:** Launches one of your ten missiles at Godzilla. The missile may be launched from any sector and will be aimed automatically.
7. **Atomic Bomb:** Launches your one atomic bomb at Godzilla. Use the bomb very carefully because the explosion will completely devastate the sector where Godzilla is plus the eight sectors surrounding him. Also, be forewarned that using the atomic bomb does not guarantee a sure win.
8. **Quit:** Stops the game before the end.

You can win only by killing Godzilla. (It can be done!) Damage to Godzilla is cumulative, so if you continue to inflict enough injury, he *will* die.

Godzilla can win in two ways: if too many people are killed in battle or if Tokyo is destroyed.

GODZILLA for the C-64 is based on a program written for the Commodore Pet by Randall Lockwood. Note: Line 62035 is not in the standard framework.

```

1 PG$="^G^O^D^Z^I^L^L^A^":AU$="BY^GLEN^FISHER":JF=1
4 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/15/83 GF
90 GOTO 62000
100 DIM TR(4,4),SH(4,4),PL(4,4),PO(4,4),LN(4,4),JR(15),JC(15),
DM(10),DM$(10)
102 POKE VIC+32,11:POKE VIC+33,0:QI=11
105 DIM BS(11),BG(11),BI(11),NT(24),DR(24)
110 SP=CRT+1016:SB=13*64
115 PRINT "{clr down right}SETTING^UP..."
120 READ T$:IF T$<>"£" THEN 120
130 R0=50:C0=24:HW=0:GW=0
140 DN$="{home down wht}":BL$="{13°space}"
150 SID=54272:FOR I=0 TO 28:POKE SID+I,0:NEXT
160 C1$="{home blu 5°down}":C2$="{home 10°down red}"
200 FOR I=0 TO 15:READ JR(I),JC(I):NEXT I
210 FOR I=1 TO 10:READ DM$(I),DM(I):NEXT I
220 FOR I=1 TO 10:READ BI(I),BS(I),BG(I):NEXT
230 FOR I=1 TO 24:READ NT(I),T:DR(I)=T*20:NEXT
510 READ N:FOR I=0 TO N-1:READ T:POKE SB+I,T:NEXT
520 FOR I=N TO 63:POKE SB+I,0:NEXT
530 A=SB+64:FOR I=A TO A+60 STEP 3:POKE I,255:POKE I+1,255:
POKE I+2,224:NEXT
540 FOR I=0 TO 3:POKE VIC+I+I+1,0:POKE SP+I,14:NEXT:POKE VIC+16,0:
POKE SP,13
550 A=SB+128:FOR I=0 TO 63:POKE A+I,0:NEXT
560 FOR I=0 TO 7:POKE A+3*I,255:NEXT:POKE SP+1,15
600 POKE VIC+21,15:POKE VIC+23,12:POKE VIC+29,12:POKE VIC+27,14
610 POKE VIC+39,5:POKE VIC+40,1:POKE VIC+41,2:POKE VIC+42,6
620 POKE VIC+2,144
900 FOR I=1 TO 7 STEP 2:POKE VIC+I,0:NEXT:PRINT "{clr home wht
down right}BUILDING^JAPAN..."
1000 RESTORE:MP=0:FOR I=0 TO 4:FOR J=0 TO 4
1010 READ A,PO:PL=0:SH=0:IF PO>0 THEN PO=INT(1E7/PO+.5)
1020 IF PO>0 THEN PL=INT(RND(1)*100)+1:TR=INT(PO/80)
1030 IF PO=0 THEN SH=INT(RND(1)*6)+1:TR=SH*350
1035 IF PL>MP THEN MP=PL
1040 PL(I,J)=PL:SH(I,J)=SH:PO(I,J)=PO:TR(I,J)=TR:LN(I,J)=(PO>0)
1045 NEXT J,I
1050 DATA 83,0, 87,0, 92,0, 97,1,102,0
1060 DATA 283,0,287,0,291,0,256,1,301,1
1070 DATA 483,0,487,0,492,0,456,1,501,0
1080 DATA 683,0,687,1.5,691,1,655,.4,701,0
1090 DATA 844,1,806,1,851,0,856,0,861,0
1400 PRINT "{clr cyn}";

```

```

1410 FOR I=1 TO 5
1420 N=5:IF I>1 THEN N=4:FOR J=1 TO 5:PRINT "{4°Y P}";:NEXT J:PRINT
1430 FOR J=1 TO N:FOR K=1 TO 5:PRINT "{4°space N}";:NEXT K:
    IF J<N OR I<5 THEN PRINT
1440 NEXT J,I:PRINT "{home}"
1500 PRINT "{home cyn 2°down}"
1510 FOR I=1 TO 20:READ T,T$:PRINT SPC(T);T$:NEXT:POKE VIC+3,186
1520 DATA 17,"{rvs-on}~{*}",17,"{rvs-on}~~",17,"{rvs-on 4°space
    rvs-off J}",16,"{rvs-on £ 4°space rvs-off *}",15,"{rvs-on £
    rvs-off 2°U *_£}",15,"{* rvs-on * down}"
1530 DATA 16,"{rvs-on}~{*}",16,"{rvs-on}~~",16,"{rvs-on}~~",15,
    "{rvs-on £}~{rvs-off £}",14,"{rvs-on £}~~",13,"{rvs-on £}~~"
1540 DATA 11,"{rvs-on 5°space rvs-off £}",11,"{rvs-on 4°space Q}",6,
    "{rvs-on £ 9°space}",4,"{rvs-on *}~~{rvs-off £U rvs-on}
    ~~{rvs-off 3°U £}"
1550 DATA 7,"{rvs-on £ rvs-off}~{rvs-on}~{rvs-off £}",3,"{rvs-on}
    ~~{rvs-off}~{rvs-on £}",4,"{rvs-on}~,4,"{£}"
1610 PRINT C2$;"{blk}":GOSUB 9000:PRINT
1620 FOR I=0 TO 7:READ Z$:PRINT TAB(26);"{yel}";CHR$(49+I);"{pur}:";
    Z$:NEXT
1630 DATA LAND~ATTACK,SEA~ATTACK,AIR~ATTACK,MOVE~TROOPS,MOVE~SHIP
1640 DATA MISSILE,ATOM~BOMB,QUIT
1700 GD=2500:NM=10:AB=0:TK=0
1900 R1=0:C1=0
2000 GOSUB 8200:IF SG THEN ON SG GOTO 6900,6850
2005 GET T$:IF T$<>"" THEN 2005
2007 CS$=C1$:CS=6:GOSUB 8000:R1=R:C1=C
2010 IF C$="" OR C$=CHR$(13) THEN 2005
2015 IF C$="Q" THEN C$="8"
2020 IF C$<"1" OR C$>"8" THEN ER$="HUH?":GOSUB 7900:GOTO 2000
2025 TD=0:SD=0:AD=0:PD=0:AR=GR:AC=GC
2030 ON ASC(C$)-ASC("1")+1 GOTO 2500,3000,3500,4000,4500,5000,5500,
    6000
2040 GOTO 2000
2050 CS$=C2$:CS=4:GOSUB 8000:R2=R:C2=C
2060 POKE VIC+CS+1,0:PRINT C2$;"{blk}":GOSUB 9000
2070 GOTO 2000
2500 IF LN(AR,AC)=0 THEN ER$="NOT LAND":GOSUB 7900:GOTO 2000
2510 IF TR(AR,AC)<=0 THEN ER$="NO TROOPS":GOSUB 7900:GOTO 2000
2520 GOSUB 7800:IF IN=0 THEN 2000
2530 IF IN>TR(AR,AC) THEN ER$="TOO MANY":GOSUB 7900:GOTO 2000
2540 TD=INT(RND(1)*IN)
2550 DM=1+INT(IN/8000+RND(1)*3):IF DM>7 THEN DM=7
2555 DR=IN*50/125000:GOSUB 2700
2560 GOTO 6500
2700 FOR I=SID TO SID+14 STEP 7
2710 POKE I,0:POKE I+1,64:POKE I+5,0:POKE I+6,198:NEXT:N=4
2720 FOR J=1 TO DR

```

```
2730 O=N:N=N+7:IF N>18 THEN N=4
2740 POKE SID+N,129
2750 FOR I=1 TO PEEK(SID+27)/4:NEXT
2760 POKE SID+O,128
2770 FOR I=1 TO 30:NEXT
2780 NEXT J
2800 FOR J=1 TO 2:O=N:N=N+7:IF N>18 THEN N=4
2810 POKE SID+O,128:POKE SID+O+2,10:POKE SID+O,129:POKE SID+O,128
2820 FOR I=1 TO 30:NEXT:NEXT J
2830 RETURN
3000 IF LN(AR,AC) THEN ER$="NOT SEA":GOSUB 7900:GOTO 2000
3010 IF SH(AR,AC)<=0 THEN ER$="NO SHIPS":GOSUB 7900:GOTO 2000
3020 IF TR(AR,AC)/SH(AR,AC)<100 THEN ER$="TOO FEW MEN":GOSUB 7900:
GOTO 2000
3030 DM=SH(AR,AC)+INT(RND(1)*4):IF DM>7 THEN DM=7
3040 TD=INT(RND(1)*TR(AR,AC)):SD=INT(TD/300)
3050 IF SD>=SH(AR,AC) THEN SD=SH(AR,AC):TD=TR(AR,AC)
3055 GOSUB 3200
3060 GOTO 6500
3200 POKE SID+0,0:POKE SID+1,12:POKE SID+5,28:POKE SID+6,252:
POKE SID+22,110
3205 POKE SID+24,31
3210 FOR J=1 TO 3+RND(1)*SH(AR,AC)*2
3220 POKE SID+23,1:POKE SID+4,129
3230 FOR I=1 TO 200:NEXT I:POKE SID+4,128
3240 FOR I=1 TO 700*RND(1):NEXT
3250 NEXT J
3260 RETURN
3500 AR=R1:AC=C1
3510 IF PL(AR,AC)<1 THEN ER$="NO PLANES":GOSUB 7900:GOTO 2000
3520 IF ABS(GR-AR)+ABS(GC-AC)>2 THEN ER$="TOO FAR":GOSUB 7900:
GOTO 2000
3530 GOSUB 7800:IF IN=0 THEN 2000
3540 IF IN>PL(AR,AC) THEN ER$="TOO MANY":GOSUB 7900:GOTO 2000
3550 DM=INT(IN*RND(1)/3)+1:IF DM>9 THEN DM=9
3560 AD=INT(IN*RND(1)):GOSUB 3700:GOTO 6500
3700 POKE SID,32:POKE SID+1,1:POKE SID+2,64
3705 POKE SID+3,0:POKE SID+5,0:POKE SID+6,240
3710 FOR A=SID+7 TO SID+14 STEP 7:POKE A,0:POKE A+1,4:POKE A+2,0:
POKE A+3,1
3720 POKE A+5,0:POKE A+6,73:NEXT A:POKE SID+14,16:POKE SID+23,0:
POKE SID+24,15
3730 POKE SID+11,65:POKE SID+18,69
3740 FOR I=1 TO 3+RND(1)*IN/MP*5
3750 POKE SID+4,64:FOR J=0 TO RND(1)*300:NEXT J
3760 POKE SID+4,65:FOR J=0 TO 200+RND(1)*500:NEXT J
3770 NEXT I:POKE SID+4,64:POKE SID+11,64:POKE SID+18,64
3780 RETURN
```

```

4000 IF LN(R1,C1)=0 THEN ER$="NOT LAND":GOSUB 7900:GOTO 2000
4010 IF TR(R1,C1)=0 THEN ER$="NO TROOPS":GOSUB 7900:GOTO 2000
4020 GOSUB 7800:IF IN=0 THEN 4120
4030 IF IN>TR(R1,C1) THEN ER$="TOO MANY":GOSUB 7900:GOTO 4120
4035 PRINT DN$;TAB(26);"MOVING";STR$(IN)
4040 PRINT TAB(26);"{down red}WHERE TO?"
4050 CS=4:CS$=C2$:GOSUB 8000:PRINT DN$;"{2°down}";TAB(26);BL$
4055 R2=R:C2=C:IF C$<>"" THEN 4120
4060 IF R1=R2 AND C1=C2 THEN 4120
4065 IF ABS(R1-R2)+ABS(C1-C2)>2 THEN ER$="TOO FAR":GOSUB 7900:
    GOTO 4120
4070 IF LN(R2,C2)=0 THEN ER$="NOT LAND":GOSUB 7900:GOTO 4120
4080 TR(R2,C2)=TR(R2,C2)+IN:TR(R1,C1)=TR(R1,C1)-IN
4090 R=R1:C=C1:PRINT C1$:GOSUB 9000
4100 R=R2:C=C2:PRINT C2$:GOSUB 9000
4110 DL=60:GOSUB 7950
4120 POKE VIC+5,0:PRINT C2$;"{blk}":GOSUB 9000:PRINT DN$;TAB(26);BL$
4130 GOTO 2000
4500 IF LN(R1,C1) THEN ER$="NOT SEA":GOSUB 7900:GOTO 2000
4510 IF SH(R1,C1)=0 THEN ER$="NO SHIPS":GOSUB 7900:GOTO 2000
4520 PRINT DN$;TAB(26);"{2°down}WHERE TO?"
4530 CS=4:CS$=C2$:GOSUB 8000:R2=R:C2=C:PRINT DN$;TAB(26);"{2°down}";
    BL$
4540 IF C$<>"" THEN 4630
4550 IF LN(R2,C2) THEN ER$="NOT SEA":GOSUB 7900:GOTO 4630
4560 IF ABS(R1-R2)+ABS(C1-C2)>2 THEN ER$="TOO FAR":GOSUB 7900:
    GOTO 4630
4565 IF R1=R2 AND C1=C2 THEN 4630
4570 T=INT(TR(R1,C1)/SH(R1,C1))
4580 SH(R1,C1)=SH(R1,C1)-1:SH(R2,C2)=SH(R2,C2)+1
4590 TR(R1,C1)=TR(R1,C1)-T:TR(R2,C2)=TR(R2,C2)+T
4600 R=R1:C=C1:PRINT C1$:GOSUB 9000
4610 R=R2:C=C2:PRINT C2$:GOSUB 9000
4620 DL=60:GOSUB 7950
4630 POKE VIC+5,0:PRINT C2$;"{blk}":GOSUB 9000:PRINT DN$;TAB(26);BL$
4640 GOTO 2000
5000 IF NM<1 THEN ER$="NO MISSILES":GOSUB 7900:GOTO 2000
5005 NM=NM-1:PRINT DN$;TAB(26);"MISSILE";10-NM:
    PRINT TAB(26);"LAUNCHED"
5010 PD=INT(RND(1)*PO(AR,AC))
5020 TD=INT(RND(1)*TR(AR,AC)):IF LN(AR,AC) THEN 5040
5030 SD=INT(TD/350):IF SD>SH(AR,AC) THEN SD=SH(AR,AC):TD=TR(AR,AC)
5040 DM=1+INT(RND(1)*9)
5050 GOSUB 5300
5060 PRINT DN$;TAB(26);BL$:PRINT TAB(26);BL$:GOTO 6500
5300 POKE SID+22,110:POKE SID+1,12:POKE SID+6,252:POKE SID+5,28
5310 POKE SID+13,76:POKE SID+24,31:POKE SID+11,17
5320 POKE SID+11,16:FOR I=255 TO 55 STEP -1

```

```
5330 POKE SID+8,I:POKE SID+8,I:NEXT
5340 POKE SID+11,0:POKE SID+23,1:POKE SID+4,129
5350 FOR I=1 TO 200:NEXT I
5360 POKE SID+4,128:RETURN
5500 IF AB THEN ER$="USED_IT":GOSUB 7900:GOTO 2000
5510 PRINT DN$;TAB(26);"{pur}REALLY?":PRINT TAB(26);"?{wht}";:
      GOSUB 60000
5520 PRINT DN$;TAB(26);BL$:PRINT TAB(26);BL$
5530 IF LEFT$(IN$,1)<>"Y" THEN ER$="ABORTED":GOSUB 7900:GOTO 2000
5535 PRINT DN$;TAB(26);"BOMB":PRINT TAB(26);"DETONATED":GOSUB 5700
5540 FOR R=AR-1 TO AR+1:IF R<0 OR R>4 THEN 5600
5550 FOR C=AC-1 TO AC+1:IF C<0 OR C>4 THEN 5590
5560 PRINT LEFT$("home 20°down"),1+5*R);TAB(5*C);"{red}";
5565 IF R=3 AND C=3 THEN POKE VIC+3,0
5570 FOR Z=1 TO 5:PRINT "{5°V 5°left}";:IF Z<5 THEN PRINT "{down}";
5580 NEXT Z:PRINT "home":TD=TD+TR(R,C):PD=PD+PO(R,C)
5585 TR(R,C)=0:PO(R,C)=0:SH(R,C)=0:PL(R,C)=0
5590 NEXT C
5600 NEXT R:R=AR:C=AC:AB=1
5605 PRINT DN$;TAB(26);BL$:PRINT TAB(26);BL$
5610 DM(10)=INT(RND(1)*1500)+800:DM=10
5620 TR(AR,AC)=TD:SH(AR,AC)=SD:PL(AR,AC)=AD:PO(AR,AC)=PD:GOTO 6500
5700 POKE SID+24,25:POKE SID+23,1:POKE SID+1,12:POKE SID+6,253:
      POKE SID+5,192
5710 POKE SID+4,129:N=1
5715 POKE VIC+6,24+GC*40:POKE VIC+7,50+GR*40:
      POKE VIC+27,PEEK(VIC+27) AND NOT 8
5720 FOR I=70 TO 255:POKE SID+22,I
5725 IF BI(N)=I THEN POKE VIC+39,BG(N):POKE VIC+42,BS(N):N=N+1
5730 FOR J=1 TO EXP((255-I)/34)*.7:NEXT
5740 NEXT I:POKE SID+24,79:POKE SID+4,128
5750 FOR I=15 TO 200:POKE SID+22,I
5760 IF BI(N)<>I THEN 5770
5765 POKE VIC+39,BG(N):J=BS(N):POKE VIC+42,J:N=N+1:IF J=16 THEN
      POKE VIC+7,0
5770 FOR J=1 TO 10:NEXT:NEXT I
5780 POKE SID+4,0:POKE VIC+42,6:POKE VIC+27,PEEK(VIC+27) OR 8
5790 POKE VIC+6,24+C1*40:POKE VIC+7,50+R1*40:RETURN
6000 POKE VIC+21,0:POKE SID+24,0
6010 PRINT "{clr blu}":POKE VIC+32,14:POKE VIC+33,6
6020 GET T$:IF T$<>"" THEN 6020
6030 END
6500 TR(AR,AC)=TR(AR,AC)-TD:SH(AR,AC)=SH(AR,AC)-SD:PL(AR,AC)=PL(AR,
      AC)-AD
6510 PO(AR,AC)=PO(AR,AC)-PD:GD=GD-DM(DM)
6520 PRINT DN$;"{blu}";TAB(26);DM$(DM);".DAMAGE":
      PRINT TAB(26);"TO GODZILLA{grn}"
```

```

6525 IF SD>0 THEN PRINT TAB(26); "{left}"; SD; LEFT$("SHIPS", 5+(SD=1));
      " _SUNK"
6530 IF TD+PD>0 THEN PRINT TAB(26); "{left}"; TD+PD; "DEAD": TK=TK+TD+PD
6535 IF AD>0 THEN PRINT TAB(26); "{left}"; AD; "DESTROYED"
6545 IF AR=R1 AND AC=C1 THEN PRINT C1$:GOSUB 9000
6550 DL=300:GOSUB 7950:PRINT DN$;
6560 FOR Z=1 TO 4:PRINT TAB(26); BL$:NEXT
6565 IF PO(3,3)<25000 THEN 6900
6570 IF TK>2E7 THEN 6850
6575 IF GD<1 THEN 6800
6580 IF DM<>7 AND DM<>8 THEN 2000
6585 GOSUB 8200:IF SG THEN ON SG GOTO 6900,6850
6590 PRINT DN$;TAB(26); "{red}GODZILLA":
      PRINT TAB(26); "ON _RAMPAGE{grn}"
6595 GOSUB 8500
6600 IF LN(GR,GC) THEN 6660
6610 IF SH(GR,GC)=0 THEN 6700
6620 PD=0:TD=INT(RND(1)*TR(GR,GC)/SH(GR,GC)):SH(GR,GC)=SH(GR,GC)-1
6630 IF SH(GR,GC)=0 THEN TD=TR(GR,GC)
6640 PRINT TAB(26); "1 _SHIP _SUNK"
6650 GOTO 6670
6660 PD=INT(RND(1)*PO(GR,GC)):TD=INT(RND(1)*TR(GR,GC))
6670 PO(GR,GC)=PO(GR,GC)-PD:TR(GR,GC)=TR(GR,GC)-TD
6680 PRINT TAB(26); "{left}"; TD+PD; "DEAD"
6690 IF GR=R1 AND GC=C1 THEN PRINT C1$:GOSUB 9000
6695 TK=TK+PD+TD
6700 DL=300:GOSUB 7950
6710 PRINT DN$;:FOR Z=1 TO 4:PRINT TAB(26); BL$:NEXT
6720 IF TK>2E7 THEN 6850
6730 GOTO 2000
6800 PRINT DN$;TAB(26); "{wht}GODZILLA":PRINT TAB(26); "IS _DEAD!"
6830 HW=HW+1:N=1:GOSUB 9500:GOTO 7000
6850 PRINT DN$;TAB(26); "{left red}"; TK; "DEAD"
6860 PRINT TAB(26); "{grn}GODZILLA":PRINT TAB(26); "WINS!!"
6870 GW=GW+1:N=10:GOSUB 9500:GOTO 7000
6900 PRINT DN$;TAB(26); "{wht}TOKYO _IS":PRINT TAB(26); "DESTROYED!"
6910 PRINT TAB(26); "{grn}GODZILLA":PRINT TAB(26); "WINS!!"
6920 GW=GW+1:N=10:GOSUB 9500:GOTO 7000
7000 DL=300:GOSUB 7950:PRINT DN$;
7010 FOR I=1 TO 4:PRINT TAB(26); BL$:NEXT
7020 PRINT C2$:TAB(26); "{down yel} _ GAMES _ WON _ {cyn}"
7030 PRINT TAB(26); "GODZILLA _"; RIGHTS(" _ "+STR$(GW),4)
7040 PRINT TAB(26); "YOU {6°space}"; RIGHTS(" _ "+STR$(HW),4)
7050 PRINT DN$;:GOSUB 9100:IF YN=0 THEN 6000
7060 GOTO 900
7500 GOSUB 61000:POKE SID+24,15:POKE SID,0:POKE SID+5,0:
      POKE SID+6,240
7510 P=2↑(1/12):C=40:D=C*P*P:E=D*P*P:G=E*P*P*P/2+1:POKE SID+4,17

```

```
7520 POKE SID+1,C:FOR I=1 TO 90:NEXT
7530 POKE SID+1,D:FOR I=1 TO 90:NEXT
7540 POKE SID+1,E:FOR I=1 TO 90:NEXT
7550 POKE SID+1,C:FOR I=1 TO 90:NEXT
7555 POKE SID+1,0:FOR I=1 TO 54:NEXT
7560 POKE SID+1,G:FOR I=1 TO 120:NEXT
7570 POKE SID+1,C:FOR I=1 TO 90:NEXT
7580 POKE SID+4,16:RETURN
7800 PRINT DN$;TAB(26);"{2°down grn}HOW MANY?"
7810 PRINT TAB(26);"?{wht}";:GOSUB 60000:IF IN$="" THEN IN=0:
      GOTO 7890
7820 FOR Z=1 TO LEN(IN$):Z$=MID$(IN$,Z,1)
7830 IF Z$<"0" OR Z$>"9" THEN ER$="BAD NUMBER":GOSUB 7900:GOTO 7890
7840 NEXT Z:IN=VAL(IN$)
7890 PRINT DN$;TAB(26);"{2°down}";BL$:PRINT TAB(26);BL$:RETURN
7900 PRINT DN$;TAB(26);BL$;"{up}"
7905 PRINT TAB(26);ER$;"{up}"
7910 DL=60:GOSUB 7950
7920 PRINT TAB(26);BL$
7930 RETURN
7950 TM=TI+DL
7960 IF TI<TM THEN 7960
7970 RETURN
8000 A=VIC+39+CS/2:TM=0
8010 GOSUB 8900
8020 JD=PEEK(JS) AND 31:GET C$:IF C$<>"" OR (JD AND 16)=0 THEN 8110
8030 IF TI>TM THEN POKE A,(PEEK(A)+8) AND 15:TM=TI+2
8040 JD=JD AND 15:IF JD=15 THEN 8020
8050 R=R+JR(JD):C=C+JC(JD)
8060 IF R<0 THEN R=4
8070 IF R>4 THEN R=0
8080 IF C<0 THEN C=4
8090 IF C>4 THEN C=0
8100 POKE A,PEEK(A) AND NOT 8:TM=TI+20:GOTO 8010
8110 POKE A,PEEK(A) AND NOT 8:RETURN
8200 R1=GR+INT(RND(1)*3-1):C1=GC+INT(RND(1)*3-1)
8210 IF R1<0 OR R1>4 OR C1<0 OR C1>4 THEN 8200
8230 GR=R1:GC=C1:POKE VIC,24+GC*40+6:POKE VIC+1,50+GR*40+10
8240 R=R1:C=C1:CS=6:CS$=C1$:GOSUB 8900
8250 GD=GD+15:IF GR<>3 OR GC<>3 THEN SG=0:RETURN
8260 PRINT DN$;TAB(26);"{red}GODZILLA IS":PRINT TAB(26);"IN TOKYO!":
      GOSUB 8400
8270 PD=INT(RND(1)*PO(3,3)/3):TD=INT(RND(1)*TR(3,3))
8280 PO(3,3)=PO(3,3)-PD:TR(3,3)=TR(3,3)-TD
8290 TK=TK+TD+PD:PRINT TAB(26);"{left grn}";TD+PD;"DEAD":GOSUB 8900
8300 DL=300:GOSUB 7950:PRINT DN$;:FOR I=1 TO 3:PRINT TAB(26);BL$:
      NEXT
8310 IF PO(3,3)<10000 THEN SG=1:RETURN
```

```

8320 IF TK>2E7 THEN SG=2:RETURN
8330 GOTO 8900
8400 POKE SID+0,0:POKE SID+5,0:POKE SID+6,240:POKE SID+24,15:
    POKE SID+23,0
8410 POKE SID+4,17:FOR I=1 TO 6
8420 FOR J=25 TO 50:POKE SID+1,J:FOR K=1 TO 1:NEXT:NEXT J
8430 FOR J=49 TO 26 STEP -1:POKE SID+1,J:FOR K=1 TO 1:NEXT:NEXT J
8440 NEXT I:POKE SID+4,F:RETURN
8500 POKE SID,0:POKE SID+6,240:POKE SID+24,7:POKE SID+23,0:
    POKE SID+4,17
8510 FOR I=1 TO 4:FOR J=1 TO 70:POKE SID+1,J:NEXT J:NEXT I:
    POKE SID+4,16
8520 RETURN
8900 POKE VIC+CS,C0+40*C:POKE VIC+CS+1,50+R*40:PRINT CS$:GOSUB 9000:
    RETURN
8999 PRINT "{home}";:END
9000 Z$="-":Z=PO(R,C):IF Z>0 THEN Z$=STR$(Z)
9005 PRINT TAB(26); "POP.~";RIGHT$(" {8° space}" +Z$,8)
9010 Z$="-":Z=TR(R,C):IF Z>0 THEN Z$=STR$(Z)
9015 PRINT TAB(26); "TROOPS.~";RIGHT$(" {6° space}" +Z$,6)
9020 Z$="-":Z=PL(R,C):IF Z>0 THEN Z$=STR$(Z)
9025 PRINT TAB(26); "PLANES.~";RIGHT$(" {6° space}" +Z$,6)
9030 Z$="-":Z=SH(R,C):IF Z>0 THEN Z$=STR$(Z)
9035 PRINT TAB(26); "SHIPS.~";RIGHT$(" {7° space}" +Z$,7)
9040 RETURN
9100 NO=0:PRINT TAB(26); "{wht}PLAY AGAIN?"
9110 TM=0:SW=1:PRINT TAB(26); "{cyn}YES~NO{up}"
9120 IF TI<TM THEN 9150
9130 PRINT TAB(26+NO*4);MID$(" {wht} {cyn}",SW,1);
    MID$(" YESNO{shift-space}",3*NO+1,3); "{up}"
9140 SW=3-SW:TM=TI+15
9150 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
9160 IF JB AND JD=15 THEN YN=(NO=0):RETURN
9170 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 9110
9180 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 9110
9190 GOTO 9120
9500 FOR A=SID TO SID+14 STEP 7:POKE A,0:POKE A+1,0:POKE A+5,0:
    POKE A+6,240
9510 NEXT:POKE SID+7,128:POKE SID+14,255:POKE SID+24,15:
    POKE SID+23,0
9520 POKE SID+4,17:POKE SID+11,17:POKE SID+18,17
9530 Z=NT(N):IF Z<0 THEN 9560
9540 POKE SID+1,Z:POKE SID+8,Z:POKE SID+15,Z
9550 FOR I=1 TO DR(N):NEXT I:N=N+1:GOTO 9520
9560 POKE SID+4,16:POKE SID+11,16:POKE SID+18,16
9570 RETURN
52500 DATA £
52510 DATA 0,0, 0, 0, 0, 0, 0, 0

```

```
52520 DATA 0,0, 1, 1, -1, 1, 0, 1
52530 DATA 0,0, 1,-1, -1,-1, 0,-1
52540 DATA 0,0, 1, 0, -1, 0, 0, 0
52550 :
52560 DATA NO,5, NO,5, LITTLE,20, SOME,100, SOME,100, GREAT,150,
MAJOR,300
52570 DATA MAJOR,300, SEVERE,400, GRAVE, 800
52580 :
52590 DATA 70,2,5, 110,8,2, 150,7,8, 190,1,7
52600 DATA 230,1,1, 19,7,1, 65,8,7, 111,2,8
52610 DATA 157,16,2, 199,16,5
52620 :
52630 DATA 48,2, 50,4, 0,0, 50,1, 53,2, 50,2, 45,2, 67,16, -1,0
52640 DATA 32,12,0,0, 32,8,0,0, 32,2,0,0, 32,14, 38,12, 36,8, 32,2,0,
0, 32,10
52650 DATA 30,4, 32,14, -1,0
52990 :
53000 DATA 57,0,0,0,0,0,112,0,0,236,0,1,255,0,1,255,0,3,240,0,3,224,
0,7,248,0,7
53001 DATA 236,Q,15,192,0,15,192,0,15,128,0,31,192,0,31,224,0,63,240,
0,126,240
53002 DATA 0,252,224,195,241,192,127,195,240
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);"{left}";:ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62035 PRINT "{down}~{pur}BASED~ON~A~PROGRAM~BY~{blk}
RANDALL~LOCKWOOD{2°up}"
```

```
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
    MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$ :PRINT "{7°down cyn 9°space}"
    (C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

RATRUN

(Uses joystick)

By C.T. Nadovich and Glen Fisher



You're finally going to get a taste of what it's like to be a rat. We're going to drop you into a maze, and then you have to find the piece of cheese.

The screen will show what's in front of you. To move forward, push the joystick forward. To turn left or right, move the joystick left or right. To turn around, pull the joystick back. When you turn, you won't move out of the square you're in; you'll just turn in place.

Your starting position in the maze will be marked by a big purple square. The cheese will be shown as a

yellow block. At the top of the screen, you'll see a number. This number shows how far away the cheese would be if there were no walls in the maze. Because of the walls, you may have to travel farther than the number indicates to get to the cheese.

If you become discouraged trying to find the cheese, press the joystick button. You'll be shown the maze from above with the start marked in purple, the cheese marked in yellow, and your path marked with little dots. Don't be too quick to give up, though. Once you do, the game is over, and a new maze is constructed.

```
1 PG$="^R^A^T^R^U^N^":AU$="BY^C.T.^NADOVICH^AND^GLEN^FISHER":JF=1
4 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/19/83 GF
90 GOTO 62000
100 PRINT CHR$(142):H=12:V=7:H1=H+1:POKE VIC+32,0:POKE VIC+33,0
101 DIM JD(15):QI=7:W=50*1024:V1=W+256
102 FOR I=0 TO 15:READ JD(I):NEXT:DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,
     8,5
103 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
     16)=0
110 I=RND(-TI):DIM MZ(V+1,H+1),HR(V+1,H+1)
130 DIM XL(4),N2(4),YL(4),XR(4),DM(4),DC$(4),DX(4),DY(4)
140 FOR I=0 TO 4:READ XL(I),N2(I),YL(I),XR(I):NEXT I
150 DATA 8,20,-4,31,8,12,1,27,12,6,5,24,15,2,8,22,17,0,10,21
160 FOR I=0 TO 3:READ DM(I),DC$(I):NEXT
170 DATA 4,"{red rvs-on}",2,"{yel rvs-on}",1,"{blu rvs-on}",8,
     "{grn rvs-on}"
180 FOR I=0 TO 3:READ DX(I),DY(I):NEXT:DATA 0,-1,1,0,0,1,-1,0
190 DN$="{home 21°down}"
200 CX=INT(RND(1)*H)+1:CY=INT(RND(1)*V)+1
210 C=CX:R=CY:S=0
220 FOR I=0 TO V+1:FOR J=0 TO H+1:MZ(I,J)=0:HR(I,J)=0:NEXT:NEXT
230 PRINT "{clr 2°down}SETTING_UP_MAZE..."
300 DA=0:N=0:AV=V1+C+R*H1:AW=W+C+R*H1:HR(R,C)=1:S=S+1:
IF S>=H*V THEN 900
```

```

410 IF C>1 THEN IF HR(R,C-1)=0 THEN DA=DA+2:N=N+1
420 IF C<H THEN IF HR(R,C+1)=0 THEN DA=DA+1:N=N+1
430 IF R>1 THEN IF HR(R-1,C)=0 THEN DA=DA+8:N=N+1
440 IF R<V THEN IF HR(R+1,C)=0 THEN DA=DA+4:N=N+1
450 N=INT(RND(1)*N)+1:Z=DA+1
460 ON Z GOTO 500,850,800,600,750,610,620,630,700,640,650,660,670,
     680,690,695
500 S=S-1
510 C=C+1:IF C>H THEN C=1:R=R+1:IF R>V THEN R=1
520 ON HR(R,C)+1 GOTO 510,300
600 ON N GOTO 800,850
610 ON N GOTO 750,850
620 ON N GOTO 750,800
630 ON N GOTO 750,800,850
640 ON N GOTO 700,850
650 ON N GOTO 700,800
660 ON N GOTO 700,800,850
670 ON N GOTO 700,750
680 ON N GOTO 700,750,850
690 ON N GOTO 700,750,800
695 ON N GOTO 700,750,800,850
700 MZ(R,C)=MZ(R,C) OR 4:R=R-1:MZ(R,C)=MZ(R,C) OR 1:GOSUB 895:
     GOTO 300
750 MZ(R,C)=MZ(R,C) OR 1:GOSUB 895:R=R+1:MZ(R,C)=MZ(R,C) OR 4:
     GOTO 300
800 MZ(R,C)=MZ(R,C) OR 8:C=C-1:MZ(R,C)=MZ(R,C) OR 2:GOSUB 895:
     GOTO 300
850 MZ(R,C)=MZ(R,C) OR 2:GOSUB 895:C=C+1:MZ(R,C)=MZ(R,C) OR 8:
     GOTO 300
895 PRINT LEFT$("{home 12°down}",R+4);TAB(C);:Z=(3 AND MZ(R,C))+1
897 PRINT MID$("~{PN@}",Z,1):RETURN
900 I=RND(1)>.5:J=RND(1):K=RND(1)>.5
920 IF I THEN X=INT(J*H)+1:Y=V+(V-1)*K
930 IF I=0 THEN Y=INT(J*V)+1:X=H+(H-1)*K
940 IF (CX-X)↑2+(CY-Y)↑2<(H↑2+V↑2)/9 THEN 900
950 SX=X:SY=Y
1000 PRINT "{clr wht}":POKE VIC+17,23:POKE VIC+22,7:POKE VIC+32,11
1010 FOR I=0 TO V+1:FOR J=0 TO H+1:HR(I,J)=0:NEXT:NEXT
1020 D=INT(RND(1)*4)
1030 T4=TI:HR(Y,X)=1
1045 Z$=MID$(STR$(INT(SQR((X-CX)↑2+(Y-CY)↑2)*10+.5)/10),2)
1046 GOSUB 7500:PRINT "{home}";TAB((38-LEN(Z$))/2);Z$
1049 IF CX=X AND CY=Y THEN 6000
1050 IF FNB(0) THEN K=TI:GOTO 6100
1055 GET IN$:IF IN$="Q" THEN PRINT "{clr 2°down}THANKS FOR PLAYING":
     END
1060 I=FNJ(0):IF I=5 THEN 1050
1070 IF FNJ(0)<>5 THEN 1070

```

```
1075 IF I=2 THEN D=D+2
1080 IF I=4 THEN D=D-1
1090 IF I=6 THEN D=D+1
1100 IF D<0 THEN D=D+4
1110 IF D>3 THEN D=D-4
1120 IF I<>8 THEN 1045
1200 AV=V1+X+Y*H1:AW=W+X+Y*H1
1201 ON D GOTO 1260,1310,1360
1210 IF Y>1 AND (MZ(Y,X) AND 4) THEN Y=Y-1:HR(Y,X)=1:GOTO 1045
1220 GOTO 1400
1260 IF X<H AND (MZ(Y,X) AND 2) THEN X=X+1:HR(Y,X)=1:GOTO 1045
1290 GOTO 1400
1310 IF Y<V AND (MZ(Y,X) AND 1) THEN Y=Y+1:HR(Y,X)=1:GOTO 1045
1340 GOTO 1400
1360 IF X>1 AND (MZ(Y,X) AND 8) THEN X=X-1:HR(Y,X)=1:GOTO 1045
1400 DN=23:GOSUB 7000
1410 FOR Z=1 TO 5:PRINT TAB(16);"{wht}NO^DOOR{up}":FOR I=1 TO 100:
NEXT
1420 PRINT TAB(16);"{pur}NO^DOOR{up}":FOR I=1 TO 100:NEXT:NEXT Z
1430 PRINT TAB(16);"{7°space up}":GOTO 1049
2000 PRINT "{clr wht} ";
2020 FOR I=1 TO H:PRINT "{3°P}";:NEXT:PRINT "{2°down}"
2100 FOR J=1 TO V:PRINT "{M up left M up left M 2°down}";:
FOR I=1 TO H
2125 K=MZ(J,I) AND 3
2130 PRINT MID$("{2°P @}^{M 3°P}^^",l+3*K,3);
2140 IF K<2 THEN PRINT "{up left M up left M 2°down}";
2200 PRINT "{up left}";
2210 IF HR(J,I)=1 THEN PRINT "{left}.";
2250 IF I=SX AND J=SY THEN PRINT "{pur left rvs-on}^rvs-off wht}";
2260 IF I=CX AND J=CY THEN PRINT "{yel left rvs-on}^rvs-off wht}";
2270 IF I=X AND J=Y THEN PRINT "{left}";MID$("↑>V<",D+1,1);
2280 PRINT "{right down}";
2290 NEXT:PRINT "{2°down}":NEXT:RETURN
3500 NO=0:PRINT "{up} ";
3510 TM=0:SW=1:PRINT TAB(10);"{wht}ANOTHER?{shift-space cyn}
YES^NO{up}"
3520 IF TI<TM THEN 3550
3530 PRINT TAB(19+NO*4);MID$("{wht cyn}",SW,1);
MID$("YESNO{shift-space}",3*NO+1,3);"{up}"
3540 SW=3-SW:TM=TI+15
3550 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
3560 IF JB AND JD=15 THEN 3600
3570 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 3510
3580 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 3510
3590 GOTO 3520
3600 PRINT TAB(19+NO*4);"{wht}";MID$("YESNO{shift-space}",3*NO+1,3);
"{up}":YN=(NO=0):RETURN
```

```

5000 IF S=0 OR S=4 THEN RETURN
5010 PRINT LEFT$(DN$,19);
5020 IF S=3 THEN PRINT TAB(20); "{3°up yel}.{wht}"
5030 IF S=2 THEN PRINT TAB(19); "{yel rvs-on up £N 2°left down P
rvs-off £ up wht}"
5040 IF S=1 THEN PRINT TAB(18); "{yel 2°down rvs-on £}..{N 4°left
down 2°Y P}..{4°left down}..{N rvs-off £ wht}"
5050 RETURN
5200 ON SH GOTO 5210,5220,5230,5240:RETURN
5210 PRINT LEFT$(DN$,14);TAB(18); "{pur 2°* wht}":RETURN
5220 PRINT LEFT$(DN$,15);TAB(18); "{pur 2°I down 2°left rvs-on 2°I
wht}":RETURN
5230 PRINT LEFT$(DN$,18);TAB(15); "{pur rvs-on £ 6°space * wht}":
RETURN
5240 PRINT LEFT$(DN$,21);TAB(12); "{pur rvs-on £ 12°space *}"
5250 PRINT TAB(11); "{rvs-on £ 14°space * wht}":RETURN
5300 ON SH GOTO 5310,5320,5330,5340:RETURN
5310 PRINT LEFT$(DN$,14);TAB(19); "{yel}.{wht}":RETURN
5320 PRINT LEFT$(DN$,15);TAB(18); "{yel DF down 2°left CV wht}":
RETURN
5330 PRINT LEFT$(DN$,18);TAB(17); "{yel rvs-on £N 2°left down P
rvs-off £ wht}":RETURN
5340 PRINT LEFT$(DN$,21);TAB(18); "{yel rvs-on £}..{N 4°left down
2°Y P}..{4°left down}..{N rvs-off £ wht}":RETURN
6000 K=TI:PRINT LEFT$(DN$,22);
6010 FOR I=1 TO 25:PRINT TAB(15) "DELICIOUS!{up}":FOR J=1 TO 50:
NEXT J
6020 PRINT TAB(15) "{10°space up}":FOR J=1 TO 50:NEXT J,I
6100 POKE VIC+32,0:PRINT "{clr}":POKE VIC+17,27:POKE VIC+22,8
6110 GOSUB 2000:GOSUB 3500
6120 IF YN THEN GOTO 200
6130 PRINT "{clr blu}":END
7000 PRINT "{home}";
7100 IF DN>0 THEN FOR Z=1 TO DN:PRINT "{down}";:NEXT
7110 RETURN
7500 TX=X:TY=Y:SH=5:HT=0:VT=0:DH=22:DF=D:DL=(D+3) AND 3:
DR=(D+1) AND 3
7510 L0=DM(DL):F0=DM(DF):R0=DM(DR)
7520 PRINT "{clr}"
7530 LD=DL:RD=DR
7600 H0=SH:SW=SH:IF SH=5 THEN H0=1:SW=8
7620 RT=38-SW-HT
7700 T=MZ(TY+DY(DL),TX+DX(DL))
7710 PRINT LEFT$(DN$,VT+1);
7720 BL$=LEFT$("10°space",SW)
7730 IF (T AND R0)=0 THEN DX=DL:GOSUB 9100:GOSUB 9200:GOSUB 9300:
GOTO 7750
7740 IF (T AND F0)=0 THEN DX=DF:GOSUB 9000:GOSUB 9200:GOSUB 9000:
PL=1:GOTO 7750

```

```

7750 T=MZ(TY+DY(DR),TX+DX(DR))
7760 PRINT LEFT$(DN$,VT+1);
7770 BL$=LEFT$("10° space",SW)
7780 IF (T AND L0)=0 THEN DX=DR:GOSUB 9150:GOSUB 9250:GOSUB 9350:
    GOTO 7800
7790 IF (T AND F0)=0 THEN DX=DF:GOSUB 9000:GOSUB 9250:GOSUB 9000:
    PR=1:GOTO 7800
7800 IF (MZ(TY,TX) AND F0)=0 THEN 7900
7810 TX=TX+DX(DF):TY=TY+DY(DF):HT=HT+SW:VT=VT+H0:SH=SH-1:DH=DH-SH-SH
7815 IF TX=SX AND TY=SY THEN GOSUB 5200
7816 IF TX=CX AND TY=CY THEN GOSUB 5300
7820 IF SH>0 THEN 7600
7830 PRINT LEFT$(DN$,VT+1);TAB(HT);DC$(DL);
7840 IF MZ(TY,TX) AND L0 THEN PRINT DC$(DF);"{rvs-off @}";:GOTO 7860
7850 PRINT "{* down left rvs-off £ up}";
7860 PRINT DC$(DR);:IF MZ(TY,TX) AND R0 THEN
    PRINT DC$(DF);"{rvs-off @}";:GOTO 7890
7870 PRINT "{£ down left rvs-off * up}";
7890 PRINT "{home wht}":RETURN
7900 BL$=LEFT$("39° space",RT-HT-SW)
7910 HT=HT+SW:PRINT LEFT$(DN$,VT+H0+1);
7920 FOR I=1 TO DH:PRINT TAB(HT);DC$(DF);BL$:NEXT
7930 PRINT "{wht home}":RETURN
8000 PRINT LEFT$(DN$,VT+1);TAB(HT);
8010 FOR I=H0 TO 1 STEP -1:PRINT DC$(D-1);TAB(HT);MID$(BL$,I):NEXT
8020 PRINT "{rvs-on 4° space}";TAB(36);".~.{left inst}."
8030 NEXT I
8040 PRINT "{rvs-on 39° space left inst}.{home}"
8050 GOTO 8050
9000 FOR I=1 TO H0:PRINT:NEXT:RETURN
9100 FOR I=H0 TO 1 STEP -1:PRINT TAB(HT);DC$(DX);MID$(BL$,I+1);
    "{*}":NEXT:RETURN
9150 FOR I=H0 TO 1 STEP -1:PRINT TAB(RT+I-1);DC$(DX);"{£}";MID$(BL$,
    I+1):NEXT
9160 RETURN
9200 IF DX=DF AND LD=DF THEN BL$="{H}"+MID$(BL$,2)
9210 FOR I=1 TO DH:PRINT TAB(HT);DC$(DX);BL$:NEXT:LD=DX:RETURN
9250 IF DX=DF AND RD=DF THEN BL$=MID$(BL$,2)+"{N}"
9260 FOR I=1 TO DH:PRINT TAB(RT);DC$(DX);BL$:NEXT:RD=DX:RETURN
9300 FOR I=1 TO H0:PRINT TAB(HT);DC$(DX);MID$(BL$,I+1);"{rvs-off
    £}":NEXT:RETURN
9350 FOR I=1 TO H0:PRINT TAB(RT+I-1);DC$(DX);"{rvs-off * rvs-on}";
    MID$(BL$,I+1):NEXT:RETURN
9500 PRINT "{clr 2°down}":FOR I=1 TO 4
9510 FOR J=1 TO 3:PRINT MID$("{red grn blu yel}",I,1);"{rvs-on
    right 24° space}":NEXT
9520 PRINT:NEXT I
9530 PRINT "{home}":FOR I=1 TO 19

```

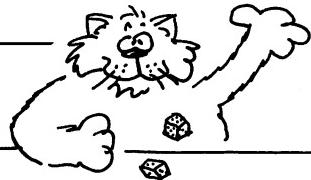
```

9540 PRINT "{rvs-on 3°right red}^^{2°right grn}^^{2°right blu}
      ^^{2°right yel}^^"
9550 NEXT I
9999 END
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);"{left}";:ZC=3-ZC:
      ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
      GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
      GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
      GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
      CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
      TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
      TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
      MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
      (C) ^1983 ^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

YAHTZEE

By Glen Fisher



YAHTZEE is a dice game with an element of skill needed. The object is to get the highest score possible by making various combinations of five dice. The game consists of thirteen rounds with each player getting one turn per round. During a turn, a player may roll the dice one, two, or three times and then score the roll in any of the thirteen categories. Any category, except Yahtzee, can be scored in only once.

When you start, the program will ask you how many players there are and what their names are. It is possible to play YAHTZEE alone, but there's no competition involved. At least two players are recommended.

To start the play, press RETURN to roll all five dice. You must type the numbers on the faces of the dice you want re-rolled. To score the roll in a category, type the letter of the category you want.

The categories are as follows:

Aces through Sixes: Six different categories that are handled almost identically. A roll of a number of 1s (or 2s or 3s and so on) with other numbers thrown in for interest. The score is the sum of all the 1s (or 2s, 3s, etc.) that you rolled. For example, a roll of three 5s is worth $5+5+5$, or 15, and is scored in the 5s category. A roll of two 1s is worth $1+1$, or 2, when scored in the 1s category. The roll 5-1-3-1-5 could get you two points in the 1s category, three points in the 3s category, or ten points in the 5s category. You choose the category to score it under.

Three of a Kind: Three of the same number (the other two dice don't matter). As an example, 5-1-5-4-5 is a Three of a Kind with three 5s. The score is the sum of all five dice, so the one above is worth $5+1+5+4+5$, or 20. Full Houses, Yahtzees, and Four of a Kind can be scored as Three of a Kind as well, since they all have three identical dice as part of the roll.

Four of a Kind: Similar to Three of a Kind, except that you need four identical dice. As with Three of a Kind, the score is the sum of all five dice. Thus, 4-4-1-4-4 would be worth $4+4+1+4+4$, or 21.

Yahtzee can be scored here also, because it has four like dice.

Full House: A Three of a Kind and a pair combined. That is, a Full House is three dice with one number plus two dice of another number. Thus, 4-5-5-4-4 is a Full House, with three 4s and two 5s. The score is 25 points, regardless of the numbers involved. A Yahtzee can be scored as a Full House. For example, three 5s plus two more 5s.

Small Straight: A combination of 1-2-3-4, 2-3-4-5, or 3-4-5-6, not necessarily in order. The fifth die is disregarded. As an example, 3-1-4-2-6 is a Small Straight. Any Small Straight is worth 30 points.

Large Straight: Either 1-2-3-4-5 or 2-3-4-5-6, possibly out of order. Thus, 3-1-5-4-2 is a Large Straight. Any Large Straight is worth 40 points.

Yahtzee: Five dice, all alike, as five 5s or five 2s. Yahtzees are worth 50 points.

Chance: Any combination of the five dice may be scored here. The point value is the sum of all five dice. This category is a safety net of sorts. A roll which is of no use elsewhere (such as 1-2-5-3-6) can be scored under Chance.

Special Rules:

Top Half Bonus: A total of 63 or more points in the Aces through Sixes category earns you a 35-point bonus. As it turns out, if you get scores from three dice in each category (three 1s, three 2s, and so on), you'll get the bonus. However, any combination that adds up to 63 or more points will get you the bonus.

Zeroing a Category: If you get a roll that can't be scored in any category that is still unused, you must zero one of the unused categories. Zeroing a category means claiming zero points in it and forfeiting any chance of improvement. Once a category has been scored it can't be changed. To zero a category, type the letter of the category to be zeroed. The C-64 will ask, **Really zero it?** to double-check. Replying with a Y will put a zero there. You can forestall the question

by typing a **Z** in front of the category, as **ZL** to zero the Yahtzee category.

Multiple Yahtzees: Yahtzee is the one category which can be scored in several times. The first Yaht-

zee you roll is worth 50 points. Any more Yahtzees you roll are each worth 100 points in the Yahtzee category. However, zeroing the Yahtzee category makes any later Yahtzees you roll worthless.

```

1 PG$="~Y~A~H~T~Z~E~E~":AU$="BY~GLEN~FISHER"
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/16/83 GF
90 GOTO 62000
100 DIM DI(5),DI$(6,6),T(5),TS(13),SC%(3,13),TL(4,3),NM$(4),SV(5):
    QI=24
105 PRINT "{clr}"
110 FOR I=1 TO 6:FOR J=1 TO 6:READ DI$(I,J):NEXT J,I
120 DATA "~~~","~{Q}~","{Q}~~~","{Q}~{Q}~","{Q}~{Q}~","{3°Q}~","~{Q}~",
    "~~~","~{Q}~","~~~","~{Q}~"
121 DATA "~~~","~~~","{Q}~~~","~~~{Q}~","{Q}~{Q}~","{Q}~{Q}~","{3°Q}~",
    "~~~","{Q}~~~","~~~{Q}~","{Q}~{Q}~"
122 DATA "{Q}~{Q}~","{Q}~{Q}~","~{Q}~","~~~","~{Q}~","~~~","~{Q}~",
    "~{Q}~{Q}~","~~~","~~~{Q}~","{Q}~"
123 DATA "{Q}~{Q}~","{Q}~{Q}~","{Q}~{Q}~"
130 DN$="{home 7°down}"
200 PRINT "HOW~MANY~PLAYERS?~";:GOSUB 60000:PRINT:NP=VAL(IN$)
210 IF NP<1 OR NP>4 THEN PRINT "ONE~TO~FOUR~PLAYERS,~PLEASE.~":
    GOTO 200
220 PRINT "{down}ENTER~THEIR~NAMES,~PLEASE.~":
    PRINT "(FIVE~LETTERS~LONG,~MAX.){down}"
240 FOR I=1 TO NP:PRINT "PLAYER" I ":";:GOSUB 60000
250 NM$(I-1)=LEFT$(IN$,20):PRINT:NEXT
300 FOR I=0 TO 3:TL(I,0)=15+6*I+5-LEN(NM$(I))
400 FOR J=1 TO 13:SC%(I,J)=-1:NEXT J:FOR J=1 TO 3:TL(I,J)=0:
    NEXT J,I
900 PL=NP:RD=0:GOSUB 10000:PRINT DN$;"{up cyn}";
920 FOR I=0 TO NP-1:PRINT TAB(TL(I,0));NM$(I);"{up}":NEXT I
1500 FOR I=1 TO 5:DI(I)=0:NEXT I:RL=0
1520 PL=PL+1:IF PL>=NP THEN PL=0:RD=RD+1:IF RD>13 THEN 5000
1530 GOSUB 11000
1540 PRINT DN$;"{home pur down right rvs-on}ROUND~";RIGHT$(STR$(RD),
    2)
1550 PRINT DN$;TAB(TL(PL,0));"{up wht}";NM$(PL)
2000 PR$=?~":GOSUB 9100:IF IN$="Q" THEN PRINT "{clr}":END
2005 IF RL=0 THEN IN$="00000"
2010 IF IN$="" THEN 2000
2020 ZER=0:IF LEFT$(IN$,1)="Z" THEN ZER=1:IN$=MID$(IN$,2):
    IF IN$="" THEN 2000
2030 N=1:A=1:FOR I=1 TO LEN(IN$)

```

```
2040 T$=MID$(IN$,I,1):IF T$<"A" OR T$>"M" THEN A=0
2050 IF T$<"0" OR T$>"6" THEN N=0
2060 NEXT I
2070 IF A THEN 3000
2080 IF N=0 THEN 2000
2090 RL=RL+1:IF RL>3 THEN PR$="NO MORE ROLLS ALLOWED.":GOTO 3500
2095 FOR I=1 TO 5:SV(I)=DI(I):NEXT
2100 FOR I=1 TO LEN(IN$):T=VAL(MID$(IN$,I,1))
2105 FOR J=1 TO 5:IF DI(J)=T THEN DI(J)=0:N=0:GOTO 2115
2110 NEXT J:PR$="NOT ENOUGH"+STR$(T)+"S":RL=RL-1
2111 FOR J=1 TO 5:DI(J)=SV(J):NEXT:GOTO 3500
2115 NEXT I:IF N THEN RL=RL-1:GOTO 3500
2117 PRINT "{home 3°down 2°right}ROLL";RL
2120 GOSUB 9000:GOSUB 8000:RV$="{rvs-on}":PRINT DN$;
2130 FOR CT=1 TO 6:GOSUB 2600:PRINT:NEXT CT
2131 PRINT:PRINT
2132 FOR CT=7 TO 13:GOSUB 2600:PRINT:NEXT CT
2133 GOTO 2000
2140 FOR I=1 TO 6:SC=TS(I):IF SC<>0 AND SC%(PL,I)<0 THEN GOSUB 2500:
   PRINT "{up}";
2150 PRINT:NEXT I:PRINT:PRINT
2160 FOR I=7 TO 13:SC=TS(I):IF SC<>0 AND SC%(PL,I)<0 THEN
   GOSUB 2500:PRINT "{up}";
2170 PRINT:NEXT I:GOTO 2000
2500 PRINT TAB(17+6*PL);RV$;RIGHT$(" . . ." +STR$(SC),3):RETURN
2510 PRINT TAB(17+6*PL);". . .":RETURN
2600 IF SC%(PL,CT)>=0 THEN RETURN
2610 SC=TS(CT)
2620 ON (SC=0)+2 GOSUB 2510,2500
2630 PRINT "{up}";:RETURN
3000 RV$="":T=ASC(IN$)-64:ADD=TS(T)
3005 IF T=12 AND ADD>0 AND SC%(PL,T)>0 THEN ADD=100:GOTO 3100
3010 IF SC%(PL,T)>=0 THEN PR$="YOU'VE ALREADY TAKEN IT.":GOTO 3500
3020 IF ADD>0 OR ZER=1 THEN 3100
3030 PR$="REALLY ZERO IT?":GOSUB 9100
3040 IF LEFT$(IN$,1)<>"Y" THEN 2000
3100 X=T:IF T>6 THEN X=X+2
3105 IF SC%(PL,T)<0 THEN SC%(PL,T)=0
3110 PRINT:PRINT DN$;LEFT$("{rvs-off 14°down}",X);
3120 RV$="":SC=ADD+SC%(PL,T):SC%(PL,T)=SC
3130 GOSUB 2500:RV$="{yel}":IF T>6 THEN 3300
3210 PRINT LEFT$("{6°down}",7-T);:X=TL(PL,1):SC=X+ADD:TL(PL,1)=SC
3220 IF X>62 OR SC<63 THEN 3250
3230 X=SC:SC=35:X=X+SC:PRINT "{up}";:RV$="{cyn}":GOSUB 2500:
   TL(PL,1)=X:RV$="{yel}"
3240 TL(PL,3)=TL(PL,3)+SC:SC=X
3250 GOSUB 2500:PRINT "{8°down}";:GOTO 3400
3300 PRINT LEFT$("{rvs-off 7°down}",14-T);
```

```

3310 SC=TL(PL,2)+ADD:TL(PL,2)=SC:GOSUB 2500
3400 SC=TL(PL,3)+ADD:TL(PL,3)=SC:RV$="{wht}":GOSUB 2500
3450 PRINT DN$;:FOR I=1 TO 6:IF SC%(PL,I)<0 THEN GOSUB 2510:
    PRINT "{up}";
3455 PRINT:NEXT I
3460 PRINT "{down}":FOR I=7 TO 13:IF SC%(PL,I)<0 THEN GOSUB 2510:
    PRINT "{up}";
3465 PRINT:NEXT I
3470 PRINT DN$;"{up cyn}";TAB(TL(PL,0));NM$(PL):GOTO 1500
3500 PRINT DN$;TAB(12);"2°up red";PR$;"{wht}":FOR I=1 TO 1000:
    NEXT I:GOTO 2000
3600 FOR I=1 TO 5:DI(I)=SV(I):NEXT:GOTO 3500
5000 TM=TI+300:FOR I=0 TO NP-1:FOR J=0 TO NP-2
5010 IF TL(J,3)>TL(J+1,3) THEN 5040
5020 T=TL(J,3):TL(J,3)=TL(J+1,3):TL(J+1,3)=T
5030 T$=NM$(J):NM$(J)=NM$(J+1):NM$(J+1)=T$
5040 NEXT J,I
5050 IF TI<TM THEN 5050
5060 PRINT "{clr wht down}FINAL_STANDINGS:{down}"
5070 FOR I=0 TO NP-1:PRINT TL(I,3);TAB(5);NM$(I):NEXT I
5100 PRINT "{down}PLAY AGAIN?~";:GOSUB 60000
5110 IF LEFT$(IN$,1)<>"N" THEN 300
5120 END
8000 FOR I=1 TO 13:TS(I)=0:NEXT I:CT=0:SUM=0
8010 FOR I=1 TO 5:T=DI(I):X=TS(T):IF X=0 THEN CT=CT+1
8020 TS(T)=X+1:SUM=SUM+T:NEXT I:TS(13)=SUM
8050 ON CT GOTO 8100,8200,8150,8300,8400
8100 TS(12)=50:TS(8)=SUM:TS(7)=SUM:GOTO 8500
8150 FOR I=1 TO 6:IF TS(I)=3 THEN TS(7)=SUM
8160 NEXT I:GOTO 8500
8200 CT=0:FOR I=1 TO 6:IF TS(I)=1 THEN CT=CT+1
8220 NEXT I:TS(7)=SUM
8240 IF CT=0 THEN TS(9)=25
8250 IF CT=1 THEN TS(8)=SUM
8260 GOTO 8500
8300 MN=7:MX=0:FOR I=1 TO 6
8310 IF TS(I)=0 THEN 8340
8320 IF I<MN THEN MN=I
8330 IF I>MX THEN MX=I
8340 NEXT I
8350 IF MX-MN=3 THEN TS(10)=30
8360 GOTO 8500
8400 IF TS(1)=0 OR TS(6)=0 THEN TS(11)=40:TS(10)=30:GOTO 8500
8410 IF TS(2)=0 OR TS(5)=0 THEN TS(10)=30:GOTO 8500
8420 GOTO 8500
8430 IF I>MX THEN MX=I
8440 NEXT I
8450 IF MX-MN=4 THEN TS(10)=30:TS(11)=40

```

```
8500 FOR I=1 TO 6:TS(I)=TS(I)*I:NEXT I
8510 RETURN
9000 PRINT "{home}";
9010 FOR I=1 TO 3:PRINT:FOR J=1 TO 5:IF DI(J)=0 THEN
    PRINT TAB(10+5*j) "^^^";
9020 NEXT J,I:FOR I=1 TO 5
9030 IF DI(I)<>0 THEN 9040
9035 Q=INT(RND(1)*6+1):T(I)=3*(RND(1)>.5)+3:CT(Q)=CT(Q)+1:DI(I)=Q
9040 NEXT I
9050 PRINT "{home wht}";:FOR I=1 TO 3:PRINT
9060 FOR J=1 TO 5:PRINT TAB(10+5*j);DIS(I+T(J),DI(J));:NEXT J,I
9070 PRINT:RETURN
9100 PRINT DN$ "{2°up 39°space up}":PRINT TAB(12);PR$;
9110 GOSUB 60000:RETURN
10000 PRINT "{clr}":GOSUB 11000
10020 T$="{yel}==TOTAL===="
10030 PRINT DN$;"{cyn}A:ACES":PRINT "B:TWOS":PRINT "C:THREES"
10040 PRINT "D:FOURS":PRINT "E:FIVES":PRINT "F:SIXES"
10050 PRINT " ^>=63 BONUS":PRINT T$
10060 PRINT "{cyn}G:3 OF A KIND":PRINT "H:4 OF A KIND":
    PRINT "I:FULL HOUSE"
10070 PRINT "J:SMALL STRAIGHT":PRINT "K:LARGE STRAIGHT"
10080 PRINT "L:YAHTZEE":PRINT "M:CHANCE"
10090 PRINT T$:PRINT "{wht}^.GRAND.TOTAL"
10100 RETURN
11000 T$="{U 3°* I}":U$="-{-}^-{-}":V$="{J 3°* K}":S$="{14°right}"
11010 PRINT "{home red}" S$T$T$T$T$:FOR I=1 TO 3
11020 PRINT S$U$U$U$U$U$:NEXT I:PRINT S$V$V$V$V$V$;"{3°up}"
11040 PRINT "{home pur rvs-on 10°space}":PRINT "{rvs-on 10°space}":
    PRINT "{rvs-on D 8°I F}"
11050 PRINT "{K 8°space rvs-on K}":PRINT "{rvs-on C rvs-off 8°I
    rvs-on V}"
11060 RETURN
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("^.+",ZC,1);"{left}"::ZC=3-ZC:
    ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT ".{left}"::
    GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}"::
    GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
    GOTO 60000
60140 GOTO 60010
```

```
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:  
CM=55296  
61010 QL=214:QI=254:RETURN  
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):  
TB=(40-LN)/2  
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)  
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"  
TAB(TB)LEFT$("{21°space}",LN)  
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$  
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN  
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"  
62050 TB=((40-LEN(MS$))/2)  
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}  
(C)^1983^THE^CODE^WORKS"  
62070 GET IN$:IF IN$<>"" THEN 62120  
62080 IF (PEEK(JS) AND 16)=0 THEN 62120  
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,  
Q)  
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K  
62110 GOTO 62070  
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

LEAP (Uses joystick)

By Glen Fisher



This is our version of the “peg-jump” puzzle. You are presented with a 4×8 board filled with pegs. Using the joystick, you throw one peg away to create a starting hole. Thenceforth, you try to remove more pegs by leaping over them. The game is over when you can’t make any more jumps. The object of the game is to leave as few pegs as possible on the screen. Leaving two pegs is a modest challenge, but leaving one is difficult indeed.

You can move a peg only by jumping over an adjacent peg into the hole on the other side. The jump must be horizontal or vertical. The peg you jumped over is removed, leaving another hole. If a peg cannot

jump over another one, then it cannot be moved at all.

To make a jump, use the joystick to move the blue cursor to the peg you want to jump with and press the button. The cursor will change to red. Move the red cursor to the hole you want to jump into, and press the button. While you're moving the red cursor, the blue cursor remains on the peg that will be doing the jumping.

You can cancel a move by placing the red cursor on top of the blue cursor and pressing the joystick button.

```

1035 PRINT TAB(HT);T$:PRINT TAB(HT);U$
1040 PRINT TAB(HT);T$
1050 PRINT TAB(HT);"Z*E*E*E*E*E*E*X""
1110 FOR I=1 TO 4:FOR J=1 TO 8:B(I,J)=1:NEXT J:NEXT I
1200 FOR C=1 TO 8:FOR R=1 TO 4
1210 GOSUB 9200
1220 NEXT R,C
1300 IF HC>0 THEN R=HR:C=HC:GOTO 1330
1310 GOSUB 2600:H=1:V=9:GOSUB 9000:PRINT "PLACE ^ THE ^ HOLE"
1320 XR=INT(RND(1)*4)+1:XC=INT(RND(1)*8)+1:SA=VIC+2:SC=VIC+40:
   GOSUB 3000
1325 HR=R:HC=C
1330 B(R,C)=0:GOSUB 9200
1340 H=-1:V=9:GOSUB 9000:PRINT "{20° space}"
1900 MV=0:TI$="000000"
2000 H=5:V=-3:GOSUB 9000:PRINT "{rvs-on}MOVE";MV+1:FR=0:FC=0
2005 GOSUB 2600:H=0:V=9:GOSUB 9000:PRINT "FROM?"
2010 SA=VIC+2:SC=VIC+40:GOSUB 3000:IF B(R,C)=0 THEN 2010
2020 FR=R:FC=C:GOSUB 9000:PRINT "{5° space}"
2050 GOSUB 2600:H=0:V=9:GOSUB 9000:PRINT "TO?"
2055 SA=VIC:SC=VIC+39:GOSUB 3000
2060 IF R=FR AND C=FC THEN POKE VIC+1,0:POKE VIC+3,0:GOTO 2000
2065 IF B(R,C)<>0 THEN 2055
2070 TR=R:TC=C:GOSUB 9000:PRINT "^^^"
2100 IF FR<>TR AND FC<>TC THEN 2900
2110 IF ABS(FR-TR)<>2 AND FC=TC THEN 2900
2120 IF ABS(FC-TC)<>2 AND FR=TR THEN 2900
2130 MR=(FR+TR)/2:MC=(FC+TC)/2
2140 IF B(MR,MC)=0 THEN 2900
2200 B(TR,TC)=B(FR,FC):B(FR,FC)=0
2210 B(MR,MC)=0
2220 R=FR:C=FC:GOSUB 9200
2230 R=TR:C=TC:GOSUB 9200
2240 R=MR:C=MC:GOSUB 9200
2250 MV=MV+1
2260 H=2:V=13:GOSUB 9000:PRINT "TIME ^";:ET$=TI$
2270 PRINT LEFT$(ET$,2);":";MID$(ET$,3,2);":";RIGHT$(ET$,2)
2280 POKE VIC+1,0:POKE VIC+3,0
2300 FOR I=1 TO 4:FOR J=1 TO 8:T=SGN(B(I,J))
2310 IF I<3 THEN IF T<>SGN(B(I+2,J)) AND B(I+1,J)<>0 THEN 2000
2320 IF J<7 THEN IF T<>SGN(B(I,J+2)) AND B(I,J+1)<>0 THEN 2000
2330 NEXT J,I
2400 H=0:V=11:GOSUB 9000:PRINT "YOU ^ HAVE ^ NO ^ MOVE. "
2410 PG=31-MV:IF PG<BP THEN BP=PG:BT$=ET$
2420 IF PG=BP AND ET$<BT$ THEN BT$=ET$
2500 H=0:V=15:GOSUB 9000:GOSUB 3500
2510 IF YN THEN 1000
2520 PRINT "{clr 2°down}YOUR ^ BEST ^ GAME ^ WAS:{6° space down}"

```

```

2530 PRINT BP;"PEG";:IF BP<>1 THEN PRINT "S";
2535 PRINT "~LEFT~IN~";
2540 IF LEFT$(ET$,2)<>"00" THEN PRINT LEFT$(ET$,2);":";
2550 PRINT MID$(BT$,3,2);":";RIGHT$(BT$,2);"."
2590 END
2600 IF (PEEK(JS) AND 16)=0 THEN 2600
2610 RETURN
2800 ET$=TI$:GOTO 2410
2900 IF ER<0 THEN 2950
2905 H=4:V=11:GOSUB 9000:PRINT "{red}BADMOVE.{wht}"
2910 TM=TI+90
2920 IF TI<TM THEN 2920
2930 GOSUB 9000:PRINT "{9°space}"
2950 POKE VIC+1,0:POKE VIC+3,0:GOTO 2000
3000 R=XR:C=XC:TM=0:ZT=0
3010 POKE SA,16*C+C0:POKE SA+1,16*R+R0
3020 IF TI>TM THEN POKE SC,(PEEK(SC)+8) AND 15:TM=TI+15
3030 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
3035 GET IN$:IF IN$="Q" THEN POKE VIC+21,0:PRINT "{clr 2°down}
THANKS FOR PLAYING":END
3040 IF JB AND JD=15 THEN POKE SC,PEEK(SC) AND 7:XR=R:XC=C:RETURN
3043 IF TI<ZT THEN 3020
3046 ZT=TI+10
3050 IF JD AND 1 THEN R=R+1:IF R>4 THEN R=1
3060 IF JD AND 2 THEN R=R-1:IF R<1 THEN R=4
3070 IF JD AND 4 THEN C=C+1:IF C>8 THEN C=1
3080 IF JD AND 8 THEN C=C-1:IF C<1 THEN C=8
3090 GOTO 3010
3500 NO=0
3510 TM=0:SW=1:PRINT TAB(H+HT);"{wht}TRY AGAIN?{shift-space cyn}
YES NO{up}"
3520 IF TI<TM THEN 3550
3530 PRINT TAB(H+HT+11+NO*4);MID$("{wht cyn}",SW,1);
MID$("YESNO{shift-space}",3*NO+1,3);"{up}"
3540 SW=3-SW:TM=TI+15
3550 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
3560 IF JB AND JD=15 THEN 3600
3570 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 3510
3580 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 3510
3590 GOTO 3520
3600 PRINT TAB(H+HT+11+NO*4);"{wht}";MID$("YESNO{shift-space}",
3*NO+1,3);"{up}":YN=(NO=0):RETURN
8999 END
9000 PRINT "{home}":POKE QL,V+VT:PRINT "{up}";TAB(HT+H);:RETURN
9200 H=2*C-1:V=2*R-1:GOSUB 9000:PRINT MID$(DT$,1+2*B(R,C),2):RETURN
9210 RETURN
9250 B=B(R,C):B(R,C)=B+3:GOSUB 9200:B(R,C)=B:RETURN
9999 END

```

```

60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1); "{left}"; :ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$(" {21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$(" {21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
(C) ^1983 ^THE ^CODE ^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$(" {pur cyn}",K+1)LEFT$(MS$,
Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

BOX (Uses joystick)

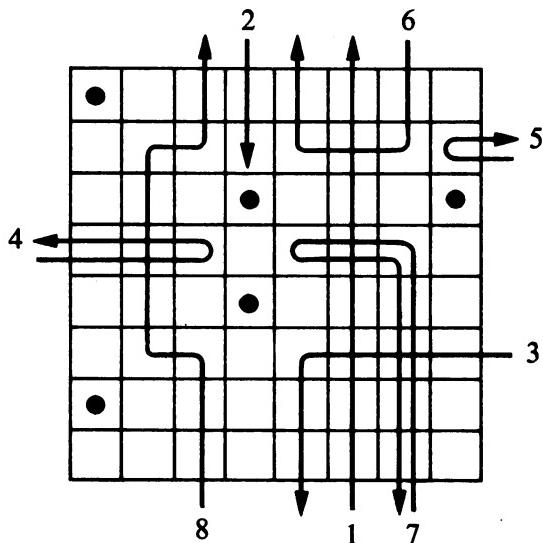
By Howard Arrington and Glen Fisher



The premise of BOX is this: you have before you a small black box containing several atoms. You must find out exactly where the atoms are in the box. Your only device to detect the atoms is a small ray generator which you can use to send rays into the box to see where, or if, the rays come out again.

The inside of the box is divided into 64 squares forming an 8×8 checkerboard pattern. Any square may have an atom in it, but a given square will never contain more than one atom. The rays will travel only along the rows and columns of the checkerboard, never diagonally.

To find the atoms, you must know how the rays interact with the atoms when you shoot them into the side of the box. (The numbers refer to the rays shown in the illustration.)



1. If a ray doesn't interact with an atom, it travels straight through the box and comes out the opposite side.
2. A ray that runs into an atom head-on scores a *hit* and never comes out.
3. A ray that passes beside an atom is *deflected* 90 degrees. It is deflected before it actually reaches the atom.

4. A ray that passes between two atoms separated by an empty square is *reflected* and returns the way it came.
5. A ray that enters beside an atom at the edge of a square is also reflected.

Rays can interact with more than one atom on their way through the box:

6. This ray is deflected by two atoms and comes out the same side it went in.

Interactions can produce unexpected results:

7. This ray is reflected. However, the deflections before and after the reflection change the apparent position of the reflecting atoms.
8. This outcome is even more deceptive. The ray is deflected by four different atoms but comes out as if it had not been deflected at all.

When you start BOX, you'll be asked how many atoms you want to search for. Change the number with the joystick, and press the button to tell BOX how many atoms you want. The screen will then be cleared, and BOX will display the 8×8 box with the possible atom locations marked with dots.

On the outside edge of the box, you'll see a small blinking arrow that you can move with the joystick. The arrow shows where the ray will enter if you fire it by pressing the joystick button. BOX will follow the ray's path and report if, or where, it comes out. Hits are marked by an H where the ray went in. Reflections are marked with an R where the ray entered. If the ray emerges anywhere else, both the entrance and the exit points are marked with a number (the same number at both ends).

To guess the location of an atom, you move the arrow into the box, and it will change to a square. Then move the square to where you think the atom might be and press the button. If you're right, the atom will appear. If you're wrong, the spot you guessed will be marked so you will not accidentally guess it again. When you find all the atoms, the game is over.

BOX keeps a running score during the game to show how well you're doing. You are awarded one point for each hit or reflection, two points for each ray that comes out elsewhere, and five points for each

wrong guess of an atom's location. As you might guess, the lower your score is, the better a player you are.

```

1 PG$="~B~O~X~":"AU$="BY~HOWARD~ARRINGTON~AND~GLEN~FISHER":JF=1
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/16/83 BS
90 GOTO 62000
100 DIM HT(32),VT(32),B(9,9),G(9,9):QI=18
101 FOR I=1 TO 8
102 HT(I)=8:VT(I)=2*I+3
103 HT(I+8)=2*I+10:VT(I+8)=22
104 HT(I+16)=29:VT(I+16)=2*(9-I)+3
105 HT(I+24)=28-2*I:VT(I+24)=1
106 NEXT I
110 GOSUB 5000
115 DN$="{home 23°down 15°right}"
125 DEF FNR(Z)=INT(8*RND(1)+1)
130 NA=5
150 GOSUB 6000:HA=NA:S=0:CX=1:CY=0
165 FOR J=0 TO 9:FOR I=0 TO 9:B(I,J)=0:G(I,J)=0:NEXT I:NEXT J
175 FOR I=1 TO NA
180 X=FNR(1):Y=FNR(1)
185 IF B(X,Y)<>0 THEN 180
190 B(X,Y)=1:NEXT I
195 GOSUB 1000:RN=0:CL=1:POKE VIC+21,1
200 PRINT "{home 2°down cyn}SCORE:";S;"{left}~~"
205 GET T$:IF T$<>"" THEN 205
210 GOSUB 4000:PRINT DN$;"{9°space}":IF Z$<>"" THEN 3500
215 IF CX>0 AND CX<9 AND CY>0 AND CY<9 THEN 3000
220 POKE VIC+39,0
230 X=CX:Y=CY:U=0:V=0:GOSUB 900
240 IF X>=1 AND X<=8 THEN V=-SGN(Y-4)
250 IF Y>=1 AND Y<=8 THEN U=-SGN(X-4)
300 X1=X+U:Y1=Y+V
310 IF U=0 THEN X2=X1-1:X3=X1+1:Y2=Y1:Y3=Y1:GOTO 330
320 Y2=Y1-1:Y3=Y1+1:X2=X1:X3=X1
330 ON 8*B(X1,Y1)+B(X2,Y2)+2*B(X3,Y3)+1 GOTO 400,410,420,410
340 S=S+1:PRINT DN$;"{red}~~HIT~~":T$="H":GOSUB 800:GOTO 200
400 X=X1:Y=Y1:GOTO 500
410 Z=1:GOTO 450
420 Z=-1
450 IF U=0 THEN U=Z:V=0:GOTO 500
460 U=0:V=Z

```

```

500 ON (X+15)/8 GOTO 550,520,560
510 STOP
520 ON (Y+15)/8 GOTO 570,300,580
530 STOP
550 Z=Y:GOTO 600
560 Z=25-Y:GOTO 600
570 Z=33-X:GOTO 600
580 Z=8+X
600 IF R=Z THEN S=S+1:PRINT DN$; "{blu}REFLECTED":T$="R":GOSUB 800:
      GOTO 200
610 S=S+2:PRINT DN$; "{grn}~EMERGED~":RN=RN+1
620 T$=MID$(STR$(RN),2):GOSUB 800
630 R=Z:T$=MID$(STR$(RN),2):GOSUB 800
640 GOTO 200
700 POKE QL,VT(R):PRINT "{up}";TAB(HT(R));C$;
710 IF LEN(T$)=1 AND R<=8 OR R>=25 THEN PRINT "{shift-space}";
720 PRINT T$:RETURN
800 POKE QL,VT(R):PRINT "{up}";TAB(HT(R));
810 IF LEN(T$)=1 AND (R<9 OR R>24) THEN T$="~"+T$
820 Z$="":IF (R>=9 AND R<=16) OR (R>=25 AND R<=32) THEN
      Z$="{left down}"
830 T$=T$+"~":"PRINT MID$(T$,1,1);Z$;MID$(T$,2,1)
840 RETURN
900 R=-1
910 IF X=0 THEN R=Y
920 IF Y=9 THEN R=X+8
930 IF X=9 THEN R=25-Y
940 IF Y=0 THEN R=33-X
950 RETURN
1000 PRINT "{clr 2°down}"
1010 PRINT TAB(11); "{wht A 15°* S}"
1020 FOR I=1 TO 8
1030 PRINT TAB(11); "- wht":~:~:~:~:~:{wht -}"
1040 IF I<8 THEN PRINT TAB(11); "- 15°space -"
1050 NEXT I
1060 PRINT TAB(11); "{wht Z 15°* X}"
1070 RETURN
1100 RETURN
3000 POKE VIC+39,0
3020 POKE QL,3+2*CY:PRINT TAB(10+2*CX); "{up}~{left}";
3025 IF G(CX,CY) THEN 3100
3030 G(CX,CY)=1
3040 IF B(CX,CY)=0 THEN S=S+5:PRINT "{wht rvs-on V}":
      PRINT DN$; "{red}**WRONG**":GOTO 200
3050 PRINT "{yel Q}":HA=HA-1:PRINT DN$; "{wht}**RIGHT**"
3060 IF HA=0 THEN PRINT DN$; "{wht 3°left}CONGRATULATIONS!":GOTO 3600
3070 GOTO 200
3100 IF B(CX,CY) THEN PRINT "{yel Q}"

```

```

3110 IF B(CX,CY)=0 THEN PRINT "{wht rvs-on V}"
3120 PRINT DN$; "{blu}^USELESS":GOTO 200
3400 PRINT DN$; "{wht 3°left}CONGRATULATIONS!"
3410 GOTO 3600
3500 FOR CY=1 TO 8:FOR CX=1 TO 8
3510 IF B(CX,CY)<>1 OR G(CX,CY) THEN 3530
3520 POKE QL,3+2*CY:PRINT TAB(10+2*CX); "{up red Q}"
3530 NEXT CX,CY
3600 POKE VIC+21,0:NO=0
3610 TM=0:SW=1:PRINT DN$; "{down grn 4°left}TRY AGAIN?{shift-space}
YES^NO{up}"
3620 IF TI<TM THEN 3650
3630 PRINT TAB(22+NO*4);MID$("{wht grn}",SW,1);
MID$("YESNO{shift-space}",3*NO+1,3); "{up}"
3640 SW=3-SW:TM=TI+15
3650 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
3660 IF JB AND JD=15 THEN 3700
3670 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 3610
3680 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 3610
3690 GOTO 3620
3700 IF NO=0 THEN 150
3710 PRINT "{clr}":END
4000 T=0:IF CX=0 THEN T=1
4010 IF CX=9 THEN T=3
4020 IF CY=0 THEN T=2
4030 IF CY=9 THEN T=4
4040 X=CX*16+94:Y=CY*16+65
4050 POKE SP,SO+T:POKE VIC,X:POKE VIC+1,Y:POKE SP,SO+T
4060 IF TI>TM THEN POKE VIC+39,CL:CL=12-CL:TM=TI+15
4100 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15:DX=0:DY=0
4105 GET Z$:IF Z$<>"" THEN RETURN
4110 IF (JD AND 1)=0 THEN DY=-1
4115 IF (JD AND 2)=0 THEN DY=1
4120 IF (JD AND 4)=0 THEN DX=-1
4125 IF (JD AND 8)=0 THEN DX=1
4130 CX=CX+DX:CY=CY+DY
4135 IF (CX<1 OR CX>8) AND (CY<1 OR CY>8) THEN CX=CX+DX+DX:
CY=CY+DY+DY
4140 IF CX>9 THEN CX=CX-10
4145 IF CX<0 THEN CX=CX+10
4150 IF CY>9 THEN CY=CY-10
4155 IF CY<0 THEN CY=CY+10
4160 IF DX OR DY THEN CL=1:TM=0:GOTO 4000
4165 IF JB=0 THEN 4000
4170 RETURN
5000 RESTORE:SB=16384-64*5:SP=CRT+1016:SO=SB/64
5010 READ T$:IF T$<>"£" THEN 5010
5020 FOR I=0 TO 4:SA=SB+64*I

```

```

5030 READ N:FOR J=0 TO N-1:READ T:POKE SA+J,T:NEXT
5040 FOR J=N TO 63:POKE SA+J,0:NEXT
5050 POKE SP+I,SO+I:NEXT I
5060 POKE VIC+1,0
5070 RETURN
6000 PRINT "{clr down wht}HOW MANY ATOMS? ^{cyn}";
6010 FOR I=1 TO 10
6030 PRINT STR$(I);:NEXT:PRINT "{up}"
6040 C=1:TM=0
6100 T=NA*2+14
6110 IF TI>TM THEN PRINT TAB(T);MID$("{wht cyn}",C,1);STR$(NA);"{up
cyn}":C=3-C:TM=TI+15
6120 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15:OA=NA
6130 IF (JD AND 8)=0 AND NA<10 THEN NA=NA+1
6140 IF (JD AND 4)=0 AND NA>1 THEN NA=NA-1
6150 IF OA<>NA THEN PRINT TAB(T);STR$(OA);"{up}":TM=0:C=1:GOTO 6100
6160 IF JB=0 THEN 6100
6170 PRINT TAB(T);"{wht}";STR$(NA):RETURN
52999 DATA F
53000 DATA 30,0,255,240,0,192,48,0,192,48,0,192,48,0,192,48,0,192,48,
0,192,48,0
53001 DATA 192,48,0,192,48,0,255,240
53010 DATA 23,0,0,0,0,0,0,3,0,0,3,128,0,63,192,0,63,192,0,3,128,0,3
53020 DATA 26,0,0,0,0,6,0,0,6,0,0,6,0,0,6,0,0,31,128,0,31,128,0,15,0,
0,6
53030 DATA 23,0,0,0,0,0,0,12,0,0,28,0,0,63,192,0,63,192,0,28,0,0,12
53040 DATA 26,0,0,0,0,6,0,0,15,0,0,31,128,0,31,128,0,6,0,0,6,0,0,6,0,
0,6
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("^{+}",ZC,1);"{left}";:ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "^{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2

```

```
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
    TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
    MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}"
    (C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
    Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

LAWN (Uses joystick)

By Kathy Higby



Summer has come again, and your lawn is in need of a trim. As a service to our customers, we offer a way you can cut your grass without leaving the comfort of your chair.

The object of LAWN is to mow the lawn of a suburban house as quickly as possible. As gasoline is

expensive, you should make an attempt to do it efficiently. If you take too long, the tank will run dry. As with any lawn, this one has occasional weed patches and rocks, both of which will stall the mower.

LAWN has a selection of skill levels that determine how quickly the tank will run dry.

```
1 PG$="~L~A~W~N~":"AU$="BY~KATHY~HIGBY":JF=1
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/16/83 GF
90 GOTO 62000
100 DIM JD(15),M(9):FOR I=0 TO 15:READ JD(I):NEXT
101 DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,8,5
102 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
16)=0
105 GOSUB 1000
107 T=WD:FOR I=2 TO 8 STEP 3:FOR J=-1 TO 1:M(I+J)=T+J:NEXT J:
T=T-WD:NEXT I
108 FOR I=0 TO 28:POKE SID+I,0:NEXT:POKE SID+6,240:POKE SID+4,17:
VL=SID+24
110 D=0:P=0:D=G/40:F=159
120 AA$="{home 17°down 20°right}"
130 X=CRT+WD*23+38:A=5:S=SID+1:POKE VL,6:POKE S,0
131 XC=X-CRT+CM
140 PRINT "{clr wht}PERCENT:{5°space}FUEL:{rvs-on 20°space
rvs-off}"
150 PRINT "{wht A 6°* F cyn 4°I wht DFD 24°* S}"
160 PRINT "{up - grn 6°+ wht G cyn rvs-on 4°space rvs-off wht HG-
grn 24°+ wht -}"
170 PRINT "{up - grn 6°+ wht G cyn rvs-on 4°space rvs-off wht HG-
grn 2°+ wht UI grn 20°+ wht -}"
180 PRINT "{up - grn 6°+ wht G cyn rvs-on 4°space rvs-off wht HG-
grn 2°+ wht JVI grn 19°+ wht -}"
190 PRINT "{up - grn 6°+ wht G cyn rvs-on 4°space rvs-off wht HG-
grn 3°+ wht JK grn 19°+ wht -}"
200 PRINT "{up - grn 5°+ wht UG cyn rvs-on 4°space rvs-off wht HG-
grn 24°+ wht -}"
210 PRINT "{up - grn 4°+ wht U+K cyn rvs-on 4°space rvs-off wht
HGJ 2°* I grn 21°+ wht -}"
```

```

220 PRINT "{up - grn 4°+ wht JV rvs-on wht G}~{M}~~{rvs-off wht
cyn 4°I wht - grn 13°+ wht U*I grn 5°+ wht - }"
230 PRINT "{up - grn 5°+ wht B rvs-on wht G}~{M}~~{rvs-off wht UI}
~{cyn rvs-on}~{rvs-off wht JI grn 12°+ wht JV+I grn 4°+ wht - }"
240 PRINT "{up - grn 5°+ wht J rvs-on wht G}~{M}~~{rvs-off wht JK
cyn rvs-on}~{rvs-off wht - grn}*{grn + grn}*{grn 2°+ grn}
*{grn + grn}*{wht UI grn 3°+ wht JV- grn 4°+ wht - }"
250 PRINT "{up - grn 6°+ rvs-on wht}~{M}~{N 15°space rvs-off wht
K grn 4°+ wht JK grn 4°+ wht - }"
260 PRINT "{up - grn 6°+ rvs-on wht}~{M}~{N 16°space rvs-off grn}
*{grn 10°+ wht - }"
270 PRINT "{up - grn 6°+ rvs-on wht}~{MN 17°@ rvs-off grn 11°+
wht - }"
280 PRINT "{up - grn 6°+ rvs-on wht}~{MM 17°space rvs-off grn
11°+ wht - }"
290 PRINT "{up - grn 6°+ rvs-on wht}~{M}~{M 16°space rvs-off grn}
*{grn 10°+ wht - }"
300 PRINT "{up - grn 5°+ wht U rvs-on wht}~{M}~{M 15°space
rvs-off grn 11°+ wht - }"
310 PRINT "{up - grn 5°+ wht JK grn 2°+ yel 6°B rvs-on wht 13°T
rvs-off grn 10°+ wht - }"
320 PRINT "{up - grn 13°+ yel 2°B wht rvs-on}~{blu 11°space wht}
~{rvs-off grn 10°+ wht - }"
330 PRINT "{up - grn 13°+ yel 2°B wht K}~{blu rvs-on 8°space wht}
~{rvs-off grn 10°+ wht - }"
340 PRINT "{up - grn 13°+ yel 2°B wht rvs-on}~{blu 11°space wht}
~{rvs-off grn 10°+ wht - }"
350 PRINT "{up - grn 13°+ yel 2°B wht rvs-on 13°space rvs-off grn
10°+ wht - }"
360 PRINT "{up - grn 38°+ wht W}"
370 PRINT "{up - grn 37°+ wht - rvs-on left inst W}"
380 PRINT "{Z 36°* EX left inst * home}"
390 Y=86
391 POKE SID+14,255:POKE SID+15,1:POKE SID+20,240
392 POKE SID+19,160:POKE SID+20,250
395 POKE SID+18,33
400 A=FNJ(0):IF A<>5 THEN IF JW<>1 THEN W=M(A)
401 ON JW+1 GOTO 404,402,403
402 JW=2:GOTO 404
403 IF A<>5 THEN POKE VL,6:POKE SID+18,33:PRINT AA$;"{rvs-on wht
5°T}":JW=0
404 GET IN$:IF IN$="Q" THEN PRINT "{clr 2°down}THANKS FOR PLAYING":
POKE SID+24,0:END
405 Z=PEEK(X+W):FR=255:IF Z=32 THEN 430
410 IF Z<>102 THEN 440
420 FR=128:P=P+1:IF RND(1)<.007 THEN 770
430 POKE X,32:X=X+W:XC=XC+W
440 POKE X,Y:POKE XC,1:POKE SID+14,FR

```

```

445 IF TI>TM THEN Y=177-Y:TM=TI+4
450 IF JW=0 THEN F=F-D:PRINT "{home wht}" TAB(8)INT(P/506*100)
455 ZZ=INT(F/8):PRINT "{home}";TAB(ZZ+18);MID$("rvs-off",{rvs-off
G rvs-off H rvs-off J rvs-off K rvs-on L rvs-on N rvs-on M},
2*INT(F-8*ZZ)+1,2)
460 REM
470 IF F>0 AND P<506 THEN 400
480 POKE SID+18,0
560 MS$="{rvs-on}OUT_OF_FUEL!":IF P=506 THEN
MS$="{rvs-on}YOU WON!{rvs-off 7°space}"
565 PRINT "{home}" MS$ "{24°space}":IF P<506 THEN 575
570 POKE X,215:FOR X=1 TO 250 STEP.5:POKE S,X:NEXT:POKE S,0:
GOTO 600
575 POKE SID+14,64:POKE SID+15,4
576 POKE SID+17,1:POKE SID+18,65:FOR J=1 TO 2000:NEXT:POKE SID+18,0
600 PRINT "{clr 3°down}" MS$
630 TP=INT(P/506*100):IF TP>BP THEN BP=TP
635 GM=GM+1:TT=TT+TP:AV=INT(TT/GM)
640 PRINT "{2°down}GAME" GM "PLAYING_AT_LEVEL" G
645 PRINT "{2°down 11°space}PERCENT_OF_LAWN_MOWED"
650 PRINT "11°space 21°T"
653 PRINT "{down}THIS GAME:" TAB(18)TP
655 IF GM=1 THEN 670
660 PRINT "{down}AVERAGE:" TAB(18)AV
665 PRINT "{down}BEST_SO_FAR:" TAB(18)BP
670 POKE VL,0:GOSUB 1100
675 IF A THEN 110
680 POKE 59466,0:POKE 59467,0:PRINT "{clr 2°home wht}":END
690 GOTO 670
770 POKE VL,15:POKE X,32:POKE SID+18,32:PRINT "{blu}";
775 X=X+W:XC=XC+W:BB$="{rvs-on}ROCKS{rvs-off}":IF RND(1)<.5 THEN
BB$="{rvs-on}WEEDS{rvs-off}"
780 PRINT AA$;"{wht rvs-on}";BB$
790 W=0:JW=1::PRINT "{wht}";:GOTO 440
1000 PRINT "{clr wht}PICK_A_SKILL_LEVEL:{down cyn}":
1010 FOR I=1 TO 10:PRINT I;:NEXT I:PRINT:G=5:T=1:TM=0
1020 X=G:IF TI>TM THEN PRINT TAB(3*G-3);MID$("wht cyn",T,1);
"up";G:T=3-T:TM=TI+5
1030 IF FNJ(0)=4 THEN IF G>1 THEN G=G-1
1040 IF FNJ(0)=6 THEN IF G<10 THEN G=G+1
1050 IF X>G THEN PRINT TAB(3*X-3);"{cyn up}";X:T=1:TM=0
1060 IF FNB(0)=0 THEN 1020
1070 RETURN
1100 PRINT "{down wht}PLAY AGAIN?,{shift-space cyn}NO.^YES":
1110 A=1:T=1:TM=0
1120 X=A:IF TI<TM THEN 1150
1130 PRINT MID$("wht cyn",T,1);TAB(13+4*A);"up";MID$("NO.YES",
3*A+1,3)

```

```

1140 T=3-T:TM=TI+10
1150 IF FNJ(0)=4 THEN A=0
1160 IF FNJ(0)=6 THEN A=1
1170 IF X<>A THEN PRINT TAB(13+4*X); "{up cyn}";MID$( "NO,YES",3*X+1,
3):T=1:TM=0
1190 IF FNB(0)=0 THEN 1120
1200 RETURN
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1); "{left}"; :ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
(C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

KALAH (Uses joystick)

By Dale Cooper



KALAH is a board game with a long history. Originally, it was played in Africa with pebbles placed in small pits dug in the ground. More recently, elegantly carved wooden boards were introduced, using the same pebbles. The current version uses injection-molded plastic with marbles. Through it all, the game has remained basically unchanged. Our high-tech computerized version of KALAH continues the tradition, but the pebbles and board are replaced with silicon chips, your Commodore 64, and spots of light on a television screen.

Each player has a row of six pits, with a seventh pit called the *kalah* at the end of the row. (The rows face each other, with a *kalah* at each end.) A move consists of taking all the pebbles from one of your six pits and "sowing" them counter-clockwise around the board. To sow the pebbles, you drop one in the next pit over, one in the pit after that, and so on around the board. In circling the board, you include your *kalah* but not your opponent's. Thus, 13 pebbles will circle the board once, with one pebble going into your *kalah* and the last pebble going into the pit the pebbles originally came from. The game is over when one player's pits are empty. The other player receives the

remaining pebbles from his row, which are transferred into his *kalah*. The winner is whoever ends up with more pebbles in his *kalah*.

The game is livened up by a couple of special rules. First, if the last pebble you drop lands in your *kalah*, you get another turn. Second, if the last pebble you drop lands in an empty pit of yours opposite an occupied pit of your opponent's, then that pebble, along with all the pebbles in the opposite pit, is moved into your *kalah*.

In our version of the game, you have the choice of playing against the C-64 or against another person. You also choose how many pebbles are in each pit to begin with. You make your move by using the joystick to move the blinking cursor to the pit you want to move from and then pressing the button. To end the game early, press the Q key when it's your turn.

If you choose to play against the C-64, be warned that it's likely to be a better player than you are. Also, the more pebbles you start with, the harder the game is and the longer it lasts.

For a good description of a KALAH program, see Charles Wetherell's book, *Etudes for Programmers* (Prentice-Hall, 1978).

```
1 PG$="^K^A^L^A^H^":AU$="BY^DALE^COOPER":JF=1
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/16/83 GF
90 GOTO 62000
100 DN$="{home 22°down}":BL$="{39°space}"
110 MD$=LEFT$(DN$,13)
120 DIM PT(14),CT(12),HT(12),KL(2),TB(2),OP(2),BR(2),PC$(2):NS=5
130 SC(1)=0:SC(2)=0:GOSUB 2200
140 PC$(1)="{yel}":PC$(2)="{cyn}"
180 PR$="WANT^TO^PLAY^AGAINST^THE^C64":GOSUB 3500:PRINT
190 IF YN THEN CH=1:PL$(1)="HUMAN":PL$(2)="C64":GOTO 260
210 CH=0
220 PRINT "{down}PLAYER^#1'S^NAME?^";:GOSUB 60000
```

```

230 PL$(1)=LEFT$(IN$,20):IF IN$="" THEN PL$(1)="FIRST PLAYER"
240 PRINT "{down}PLAYER #2'S NAME?";:GOSUB 60000
250 PL$(2)=LEFT$(IN$,20):IF IN$="" THEN PL$(2)="SECOND PLAYER"
260 TB(1)=INT(20-LEN(PL$(1))/2):OP(1)=1:BR(1)=11
265 TB(2)=INT(20-LEN(PL$(2))/2):OP(2)=1:BR(2)=11
270 GOSUB 4000
290 KL(1)=7:KL(2)=14
300 FOR PT=1 TO 14:PT(PT)=NS:NEXT:PT(KL(1))=0:PT(KL(2))=0
310 GOSUB 820:NP=30
315 FOR PL=1 TO 2:PT=KL(PL):GOSUB 2000:NEXT
320 PL=1:FOR I=1 TO 8 STEP 7:FOR PT=I TO I+5:GOSUB 2000:NEXT PT:
    PL=2:NEXT I
330 PL=1
340 RV$="{wht}":GOSUB 1800
350 IF CH=0 OR PL=1 THEN 380
355 GOSUB 1010:PT=PT+1
360 GOSUB 1850:PRINT "{4°down}";BL$;"{up}"
365 T=14-PT:POKE VIC+2,C0+40*T:POKE VIC+40,1
375 GOTO 410
380 GOSUB 2300:IF PT<0 THEN PT(KL(1))=0:PT(KL(2))=0:GOTO 750
390 T=PT:IF PL=2 THEN PT=14-PT
400 IF PT(PT)=0 THEN 350
410 GOSUB 1850:PRINT MID$("4°up 4°down",4*PL-3,4);BL$;"{up}"
412 PRINT TAB(5*T);PC$(PL);"{shift-space}(";MID$(STR$(PT(PT)),2);
    ")
415 GOSUB 1850:PRINT BL$:GOSUB 1530
420 IF P1=0 OR P2=0 THEN 670
430 IF PT<>KL(PL) THEN 500
440 GOSUB 1850:PRINT TAB(10);:IF PL=2 AND CH THEN PRINT "{pur}.I";
    GOTO 460
450 PRINT "{pur}YOU";
460 PRINT "~GET ANOTHER MOVE.":GOTO 340
500 RV$="{cyn}":GOSUB 1800:POKE VIC+38+PL,12
510 PL=3-PL:POKE VIC+38+PL,BR(PL):BR(PL)=1:GOTO 340
670 REM
680 PT=1:PL=1:NP=35
690 FOR PL=1 TO 2:Z=(PL-1)*7+1:K=KL(PL)
700 FOR PT=Z TO Z+5:IF PT(PT)=0 THEN 710
705 PT(K)=PT(K)+PT(PT):PT(PT)=0:GOSUB 2000:GOSUB 1900
710 NEXT PT:NEXT PL:PL=2:GOSUB 1850:PRINT "{2°up}";BL$:PRINT BL$
720 IF PT(KL(1))<>PT(KL(2)) THEN 730
725 RV$="{cyn}":PL=1:GOSUB 1800:PL=2:GOSUB 1800
727 PRINT MD$;TAB(15);"{wht}TIE GAME!":GOTO 750
730 PL=2:IF PT(KL(2))>PT(KL(1)) THEN PL=1
735 RV$="{cyn}":GOSUB 1800:PL=3-PL:RV$="{wht}":GOSUB 1800
740 PRINT MD$;TAB((32-LEN(PL$(PL))/2);"{cyn}WINNER:{wht}";PL$(PL)
750 POKE VIC+21,0:PRINT "{home}";TAB(11);"{grn}
    CURRENT STANDINGS{cyn}"

```

```

760 FOR I=1 TO 2:SC(I)=SC(I)+PT(KL(I))
765 PRINT TAB(20-LEN(PL$(I))):PL$(I);";";SC(I):NEXT
770 PRINT DN$;BL$;"{up}":PR$="PLAY AGAIN":GOSUB 3500:IF YN=0 THEN
900
780 PRINT "{clr}":PR$="SAME PLAYERS":GOSUB 3500:IF YN=0 THEN 180
800 PRINT BL$;"{up}":PR$="STILL"+STR$(NS)+"_STONE":IF NS<>1 THEN
PR$=PR$+"S"
805 PR$=PR$+"_PER_PIT":GOSUB 3500:IF YN=0 THEN 270
810 GOTO 290
820 POKE VIC+21,0:POKE VIC+39,11:POKE VIC+40,11:PRINT "{clr 6°down
cyn}"
830 T$="{A 3°* S}":B$="{Z 3°* X}":P$="~~~{QW}":K$="{-}~~{-} left
inst}~"
833 PRINT "{A 3°* S}" T$T$T$T$T$T$ " {A 2°* S left inst *}"
836 FOR I=1 TO 3:PRINT "{-}" P$P$P$P$P$P$;"~~{-}" K$:NEXT I
840 PRINT K$ "{right}" B$B$B$B$B$B$K$
843 PRINT K$TAB(35)K$
846 PRINT K$ "{right}" T$T$T$T$T$T$K$
850 FOR I=1 TO 3:PRINT K$ "{right -}" P$P$P$P$P$P$ "~~{-} left
inst}":NEXT I
853 PRINT "{Z 3°* X}" B$B$B$B$B$B$ " {Z 2°* X left inst * wht}"
860 RV$="{cyn}":FOR PL=1 TO 2:GOSUB 1800:NEXT PL
863 POKE VIC+21,3:RETURN
900 POKE VIC+21,0:PRINT "{clr 3°down}";TAB(12);"{grn}
FINAL_STANDINGS{cyn}"
910 FOR I=1 TO 2:PRINT TAB(20-LEN(PL$(I))):PL$(I);";";SC(I):NEXT
920 PRINT "{blu}":END
1010 PT=12:SC=-100000:SW=1
1015 PRINT MD$;TAB(19);
1020 GOSUB 1210:IF CT(PT)-13+PT=13*INT((CT(PT)-13+PT)/13) THEN
RETURN
1030 PT=PT-1:IF PT>6 THEN 1020
1040 CP=7:PT=7
1050 CT=CP:POKE VIC+2,C0+40*(13-CT):POKE VIC+40,1:GOSUB 1210:
GOSUB 1290
1060 ON CD GOTO 1260,1070,1180,1180
1070 PL=1:HP=1:T6=-10000:W=1
1080 I=6
1090 IF CT(I)-7+I=INT((CT(I)-7+I)/13)*13 THEN CT=I:GOSUB 1290:
GOTO 1080
1100 I=I-1:IF I>0 THEN 1090
1110 GOSUB 1510:CT=1
1120 POKE VIC,C0+40*CT:GOSUB 1290:ON CD GOTO 1140,1130,1130,1280
1130 IF CA(1)>T6 THEN T6=CA(1):T7=CA(2)-CA(1)
1140 HP=HP+1:IF HP<7 THEN CT=HP:GOSUB 1520:GOTO 1120
1150 W=W+1:IF W=2 THEN HP=1:CT=CP:PL=2:GOSUB 1210:GOSUB 1290:PL=1:
GOTO 1110
1160 GOSUB 1520:IF T7>SC-INT(2*RND(4)) THEN SC=T7:PT=CP

```

```

1170 GOTO 1260
1180 GOSUB 1240
1190 IF CA(2)-CA(1)>SC THEN SC=CA(2)-CA(1):PT=CP
1200 GOTO 1260
1210 FOR I=1 TO 6:CT(I)=PT(I):CT(I+6)=PT(I+7):NEXT I
1220 CA(1)=PT(KL(1)):CA(2)=PT(KL(2))
1230 RETURN
1240 CA(1)=CA(1)+CT(1)+CT(2)+CT(3)+CT(4)+CT(5)+CT(6)
1250 CA(2)=CA(2)+CT(7)+CT(8)+CT(9)+CT(10)+CT(11)+CT(12):RETURN
1260 CP=CP+1:PL=2:IF CP<13 THEN 1050
1270 RETURN
1280 GOSUB 1240:GOTO 1130
1290 IF CT(CT)=0 THEN CD=1:RETURN
1300 NP=CT(CT):CT(CT)=0:CD=2
1310 CT=1+CT:IF CT=7 THEN 1440
1320 IF CT=13 THEN 1490
1330 CT(CT)=CT(CT)+1
1340 NP=NP-1:IF NP>0 THEN 1310
1350 IF CT<7 AND PL=1 THEN 1400
1360 IF CT>6 AND PL=2 THEN 1400
1370 IF CT(1)+CT(2)+CT(3)+CT(4)+CT(5)+CT(6)=0 THEN CD=4:RETURN
1380 IF CT(7)+CT(8)+CT(9)+CT(10)+CT(11)+CT(12)=0 THEN CD=4:RETURN
1390 RETURN
1400 IF CT(13-CT)=0 OR CT(CT)>1 THEN RETURN
1410 CA(PL)=CA(PL)+1+CT(13-CT)
1420 CT(CT)=0:CT(13-CT)=0:CT=13-CT
1430 GOTO 1370
1440 IF PL=2 THEN 1330
1450 NP=NP-1:CA(PL)=CA(PL)+1
1460 IF NP=0 THEN CD=3:GOTO 1370
1470 IF CT=7 THEN 1330
1480 CT=1:GOTO 1330
1490 IF PL=1 THEN 1480
1500 GOTO 1450
1510 HA(1)=CA(1):HA(2)=CA(2):FOR I=1 TO 12:HT(I)=CT(I):NEXT:RETURN
1520 CA(1)=HA(1):CA(2)=HA(2):FOR I=1 TO 12:CT(I)=HT(I):NEXT:RETURN
1530 NP=PT(PT):PT(PT)=0:GOSUB 2000
1540 IF NP=0 THEN RETURN
1550 PT=PT+1:IF PT>14 THEN PT=1
1560 IF PT=KL(3-PL) THEN 1550
1570 PT(PT)=PT(PT)+1:NP=NP-1
1580 IF PT=KL(PL) THEN GOSUB 2000:GOTO 1600
1590 GOSUB 2000
1600 IF NP>0 THEN 1550
1610 IF PT=KL(1) OR PT=KL(2) THEN 1670
1620 IF INT(PT/7)+1<>PL OR PT(PT)<>1 OR PT(14-PT)=0 THEN 1670
1630 PT(KL(PL))=PT(KL(PL))+PT(PT)+PT(14-PT)
1640 PT(PT)=0:GOSUB 2000

```

```
1650 PT=14-PT:PT(PT)=0:GOSUB 2000
1660 GOSUB 1900
1670 P1=0:P2=0
1680 FOR Z=1 TO 6
1690 P1=P1+PT(Z):P2=P2+PT(Z+7)
1700 NEXT Z
1710 RETURN
1800 PRINT "{home 3°down}";MID$( "{17°down}", 
16*PL-15)TAB(TB(PL))RV$PL$(PL)
1810 RETURN
1850 PRINT "{home down}";MID$( "{21°down}", 20*PL-19);:RETURN
1900 SV=PT:PT=KL(PL):GOSUB 2000:PT=SV:RETURN
2000 PK=0:PRINT "{home 9°down yel}"
2010 T=5*PT+1:IF PT>7 THEN T=5*(14-PT)+1:PRINT "{cyn}";
2020 IF PT=KL(1) THEN PK=1:T=36
2030 IF PT=KL(2) THEN PK=1:T=1
2040 IF PK THEN PRINT "{5°down}"
2050 IF PT>0 AND PT<7 THEN PRINT "{5°down}"
2060 P$=MID$(STR$(PT(PT)),2):IF PT(PT)=0 THEN P$="^^^"
2070 IF LEN(P$)<>3 THEN P$="^"+P$:IF LEN(P$)<>3 THEN P$=P$+"^"
2090 PH=3:PV=2:IF PK THEN PV=8
2100 PU$=MID$( "{9°up 3°left}", 10-PV, PH+PV):PD$=LEFT$("right
9°down", 1+PV)
2110 PRINT TAB(T);:P2=0:PP=PT(PT):PM=PH*PV:IF PP>8*PM THEN PP=8*PM
2120 IF PP>8 THEN PRINT "{rvs-on}^{left up}";:PP=PP-8:GOSUB 2160:
GOTO 2120
2130 IF PP>0 THEN PRINT MID$( "{rvs-off @ rvs-off P rvs-off O
rvs-off I rvs-on U rvs-on Y rvs-on T rvs-on}^", 2*PP-1,2);
"{left up}";:GOSUB 2160
2140 IF PM>0 THEN PRINT "{rvs-off}^{left up}";:GOSUB 2160:GOTO 2140
2150 PRINT PU$;"{rvs-off}";P$;"{pur}":RETURN
2160 PM=PM-1:P2=P2+1:IF P2>=PV THEN PRINT PD$;:P2=0:PH=PH-1
2170 RETURN
2200 SB=13*64:SP=CRT+1016:R0=109:C0=15
2210 FOR I=SB TO SB+63:POKE I,0:NEXT
2220 POKE SB,3:POKE SB+1,255:POKE SB+2,254
2225 POKE SB+48,3:POKE SB+49,255:POKE SB+50,254
2230 FOR I=1 TO 15:A=SB+3*I:POKE A,2:POKE A+2,2:NEXT
2240 POKE SP,13:POKE SP+1,13
2250 POKE VIC,C0+40:POKE VIC+1,R0+48:POKE VIC+2,C0+40:POKE VIC+3,R0
2260 POKE VIC+23,3:POKE VIC+29,3:POKE VIC+16,0:POKE VIC+27,0
2270 POKE VIC+39,11:POKE VIC+40,11:RETURN
2300 V=VIC+2*(PL-1)
2305 PT=OP(PL):CL=11:TM=0
2310 POKE V,C0+40*PT
2320 IF TI>TM THEN CL=12-CL:POKE VIC+38+PL,CL:TM=TI+15
2325 GET Q$:IF Q$="Q" THEN PT=-1:RETURN
2330 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
```

```

2340 IF JD=15 AND JB THEN 2390
2350 IF (JD AND 4)=0 THEN PT=PT-1:IF PT<1 THEN PT=6
2360 IF (JD AND 8)=0 THEN PT=PT+1:IF PT>6 THEN PT=1
2370 IF (JD AND 12)<>12 THEN TM=0:CL=11:GOTO 2310
2380 GOTO 2320
2390 OP(PL)=PT:POKE VIC+38+PL,1:RETURN
3500 NO=0
3510 TM=0:SW=1:PRINT "{wht}" PR$ "?{shift-space cyn}YES^NO{up}"
3520 IF TI<TM THEN 3550
3530 PRINT TAB(LEN(PR$)+2+NO*4);MID$("{wht cyn}",SW,1);
    MID$("YESNO{shift-space}",3*NO+1,3);"{up}"
3540 SW=3-SW:TM=TI+15
3550 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
3560 IF JB AND JD=15 THEN 3600
3570 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 3510
3580 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 3510
3590 GOTO 3520
3600 PRINT TAB(LEN(PR$)+2+NO*4); "{wht}";MID$("YESNO{shift-space}",
    3*NO+1,3); "{up}":YN=(NO=0)
3610 JB=PEEK(JS) AND 16:IF JB=0 THEN 3610
3620 RETURN
4000 PRINT "{clr down wht}HOW^MANY^STONES^PER^PIT?^{\u00c3n down}"
4010 FOR I=1 TO 15
4030 PRINT STR$(I);:NEXT:PRINT "{up}"
4040 C=1:TM=0
4100 T=NS*2-2-(NS>9)*(NS-10)
4110 IF TI>TM THEN PRINT TAB(T);MID$("{wht cyn}",C,1);STR$(NS);"{up
    cyn}":C=3-C:TM=TI+15
4120 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15:OS=NS
4130 IF (JD AND 8)=0 AND NS<15 THEN NS=NS+1
4140 IF (JD AND 4)=0 AND NS>1 THEN NS=NS-1
4150 IF OS<>NS THEN PRINT TAB(T);STR$(OS);"{up}":TM=0:C=1:GOTO 4100
4160 IF JB=0 THEN 4100
4170 PRINT TAB(T); "{wht}";STR$(NS):RETURN
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("\u00a6\u005f",ZC,1);"\u00a6{left}";:ZC=3-ZC:
    ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "\u00a6{left}";:
    GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "\u00a6{left}";:
    GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
    GOTO 60000

```

```
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
    CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
    TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
    TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
    MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}"
    (C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
    Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

BONZO

By Randall Lockwood



This game is one that nearly everyone should be familiar with. It has appeared many times under many names. (Ours is named for the hit of the 1930s, the Bonzo Dog Chase Game.) Basically, it's a race game for two people to see who can reach the top of the board first. You may play against the C-64 or include it as a third player.

With each turn, you roll a die to tell how far you

can move. (In this version, you press a key to stop the die from rolling.) The board has some ladders and slides scattered over its checkerboard pattern. If you land on a square with a ladder, you can climb toward the top of the board. If you land on a slide, however, you fall down to the bottom of the board and must traverse that distance again. As you can tell, it's pure luck, but entertaining nonetheless.

```
1 PG$="~B~O~N~Z~O~":"AU$="BY~RANDALL~LOCKWOOD"
4 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/16/83 GF
90 GOTO 62000
100 DIM O(9),L(9),L$(0):QI=15
103 FOR I=0 TO 28:POKE SID+I,0:NEXT:POKE SID+6,240:POKE SID+4,17
110 PRINT "THERE~WILL~BE~A~SHORT~DELAY.~":
PRINT "{down}PLEASE~BE~PATIENT.~"
120 I=RND(-TI)
140 POKE SID+24,15:SN=1E20:FQ=SID+1
150 DIM A(3),P(91),Z%(91),M(3),S(3),W(3),X(3)
160 M(1)=209:M(2)=218:M(3)=144
165 X(1)=0:X(2)=WD:X(3)=WD+1:W(1)=71:W(2)=45:W(3)=30
170 FOR I=0 TO 12
180 FOR J=0 TO 6 STEP 2:P(J*13+I+1)=CRT+WD*19+1-WD*3*j+3*I:NEXT J
190 FOR J=1 TO 5 STEP 2:P(J*13+I+1)=CRT+WD*19+37-WD*3*j-3*I:NEXT J
200 NEXT I
210 VT=22
220 BL$="{up 39°space up}"
230 READ T$:IF T$<>"=" THEN 230
240 FOR I=826 TO 848:READ N:POKE I,N:NEXT
250 FOR I=1 TO 9:READ L(I):NEXT I:O(1)=-WD:O(2)=-WD:O(3)=-WD+1:
O(4)=-WD+1
255 O(5)=-WD-1:O(6)=WD+1:O(7)=WD+1:O(8)=WD-1:O(9)=WD-1
260 FOR I=0 TO 9:READ D$(I):NEXT I
270 FOR I=1 TO 9:READ D(I),LN(I):NEXT I
300 PRINT "{clr 4°down}HOW~MANY~{rvs-on}PEOPLE{rvs-off}~
~ARE~PLAYING?":PRINT "{down}(1~OR~2)?~";
305 GOSUB 60000:IF IN$="" THEN IN$="1":PRINT "{up}";TAB(10);IN$
```

```

310 NP=VAL(IN$):IF NP<1 OR NP>2 THEN 300
320 FOR I=1 TO NP
330 PRINT "{down}WHAT'S A PLAYER";I;"{left}"S NAME?";:GOSUB 60000:
N$(I)=IN$
333 IF IN$="" AND NP=1 THEN N$(I)="HUMAN":PRINT "{up}";TAB(24);
N$(I):GOTO 340
335 IF IN$="" THEN
PRINT "{down}WITH TWO PLAYERS, NAMES ARE NEEDED. ":GOTO 330
340 NEXT I:N$(3)="C64"
350 PRINT "{down}CAN I PLAY, TOO? ";:GOSUB 60000:
IF LEFT$(IN$,1)="N" THEN 400
355 IF IN$="" THEN PRINT TAB(17);"{up}YES":FOR I=1 TO 500:NEXT
390 NP=NP+1:N$(NP)="C64":M(NP)=131
400 I1$="{rvs-on blk left} {left}":I2$="{inst} {down}"
410 C1$="{blk}":B1$=C1$+"{rvs-on} {down 3°left} {down 3°left}"
" {rvs-off 2°up}":D1$=I1$+C1$+I2$:D1$=D1$+D1$+D1$
420 C2$="{red}":B2$=C2$+"{rvs-on} {down 3°left} {down 3°left}"
" {rvs-off 2°up}":D2$=I1$+C2$+I2$:D2$=D2$+D2$+D2$
515 FOR I=1 TO 10:E(I)=0:NEXT I:FOR J=1 TO 91:Z%(J)=0:NEXT J:
POKE VIC+33,0
520 PRINT "{clr}":FOR J=1 TO 3:FOR I=1 TO 6:PRINT B1$;B2$;:NEXT:
PRINT B1$;D1$;"{up}"
530 FOR I=1 TO 6:PRINT B2$;B1$;:NEXT:PRINT B2$;D2$;"{up}":NEXT J
540 FOR I=1 TO 6:PRINT B1$;B2$;:NEXT:PRINT B1$;D1$;"{up}";
541 FOR I=1 TO 4:PRINT:PRINT "{rvs-on blk 39°space left inst} ";
542 NEXT:PRINT "{home}"
545 POKE VIC+33,1
550 POKE QL,19:PRINT C1$;"{up rvs-on}->":PRINT "{home}";TAB(36);
" {rvs-on}***{down 3°left}*{V}*{down 3°left}***"
560 RESTORE:Z=INT(RND(1)*4)+1
570 IF Z>1 THEN FOR K=1 TO Z-1:FOR J=1 TO 9:READ T$:NEXT J,K
580 FOR K=1 TO 9:READ I$:IF I$<>"" THEN GOSUB 1080
585 NEXT K
590 DATA 1037681,2124167,,3103462,6111216,1098769,4025123,,2057450
600 DATA ,4032450,3064858,5131139,,5083418,1088662,1129068,1038153
610 DATA 5091835,1015379,4123741,3052957,6040523,1098769,1118965,
4074632
620 DATA 1088658,1067384,4101743,,5050331,,6072008,1038157,3136541,
2097046
650 :
710 FOR K=1 TO NP:S(K)=1:A(K)=160:POKE P(1)+X(K),M(K):NEXT K
720 FOR K=1 TO NP
730 SK=S(K):MK=M(K):XK=X(K):PSK=P(SK)
740 GOSUB 770
750 GOSUB 910
760 NEXT K:GOTO 720
770 D=INT(RND(1)*6)+1:POKE QL,VT+1
780 PRINT TAB(3);"{up blk rvs-on}";D$(0);"{up}":PRINT "{rvs-on}";

```

```

BL$  

785 POKE CRT+WD*23+7,M(K):PRINT TAB(7); "{rvs-on right}";N$(K);  

787 IF N$(K)="C64" THEN PRINT:GOTO 830  

790 PRINT "~(PRESS ~{rvs-off}RETURN{rvs-on})"  

795 GET T$:IF T$<>"" THEN 795  

800 GOSUB 880  

810 GET T$:IF T$="" THEN 800  

815 IF T$="Q" THEN 1070  

820 POKE QL,VT+2:PRINT TAB(10+LEN(N$(K)));"{up blk rvs-on  

14°space}"  

825 GOTO 860  

830 Q=INT(RND(1)*10)+3  

835 FOR J=1 TO Q  

840 GOSUB 880  

850 NEXT J  

860 Z=D:GOSUB 890:FOR I=1 TO 1000:NEXT I  

870 RETURN  

880 Z=INT(RND(1)*10)+1  

890 POKE QL,VT+1:PRINT TAB(3); "{up blu rvs-on}";D$(Z); "{up}"  

900 FOR I=1 TO 100:NEXT I:RETURN  

910 POKE PSK+XK,A(K):FOR I=1 TO D  

920 IF SK+I>=91 THEN 1000  

930 A(K)=PEEK(P(SK+I)+XK)  

940 POKE P(SK+I)+XK,M(K):POKE FQ,W(K):FOR J=1 TO 50:NEXT J:  

    POKE FQ,0  

950 IF I<D THEN POKE P(SK+I)+XK,A(K)  

960 NEXT I  

970 POKE FQ,0:SK=S(K)+D:S(K)=SK:PSK=P(SK)  

980 U=Z%(SK):IF U=0 THEN RETURN  

990 GOTO 1270  

1000 POKE QL,VT+1:PRINT TAB(3); "{up blk rvs-on}";D$(0); "{up}":  

    PRINT "{rvs-on}";BL$  

1010 PRINT TAB(5); "{up rvs-on}";N$(K); "~WINS!! {down}"  

1040 PRINT TAB(5); "{rvs-on}PLAY~AGAIN";  

1050 GOSUB 3000  

1060 IF LEFT$(IN$,1)<>"N" THEN PRINT "YES{home}":GOTO 515  

1070 PRINT "NO{home}":POKE SID+1,0:POKE VIC+33,0:PRINT "{clr wht}":  

    END  

1080 Y=VAL(LEFT$(I$,1)):X=VAL(MID$(I$,2,2))  

1090 Z=K:S=VAL(MID$(I$,4,2)):E(Z)=VAL(RIGHT$(I$,2)):Z%(S)=Z  

1100 P=CRT+120*(Y-1)+3*(X-1)  

1110 LN=LN(Z):ON D(Z) GOSUB 2000,2100,2200,2300,2400:RETURN  

1270 O=O(U):L=L(U)  

1280 LP=120:HP=60:Z=-1:IF U<=5 THEN LP=60:HP=120:Z=1  

1290 IV=INT((HP-LP)/L+.5)  

1300 S1=PSK+X(K):S2=A(K):FOR I=1 TO L  

1310 POKE S1,S2:S1=S1+O:S2=PEEK(S1):POKE S1,M(K)  

1320 FOR J=LP TO LP+IV STEP Z:POKE FQ,J:NEXT:LP=LP+IV

```

```
1330 NEXT I
1340 POKE FQ,0
1350 SK=E(U):S(K)=SK:A(K)=S2
1360 RETURN
1370 DATA ,162,4,160,0,132,251,134,252,177,251,73,128,145,251,200
1390 DATA 208,247,232,224,8,208,240,96,3,6,3,6,3,3,3,6,3,6
1430 DATA " ``{3°left down} ``{3°left down} ``"
1440 DATA " ``{3°left down} {Q} ``{3°left down} ``"
1450 DATA " {Q} ``{3°left down} ``{3°left down} ``{Q}"
1460 DATA " ``{Q} 3°left down} ``{Q} ``{3°left down Q} ``"
1470 DATA " {Q} ``{Q} 3°left down} ``{3°left down Q} ``{Q}"
1480 DATA " {Q} ``{Q} 3°left down} ``{Q} ``{3°left down Q} ``{Q}"
1490 DATA " {3°Q} 3°left down} ``{3°left down 3°Q}"
1500 DATA " ``{Q} 3°left down} ``{3°left down Q} ``"
1510 DATA " {Q} ``{3°left down} ``{Q} ``{3°left down} ``{Q}"
1520 DATA " {Q} ``{Q} 3°left down Q shift-space Q 3°left down Q
shift-space Q}"
1600 DATA 2,4, 2,7, 3,2, 3,5, 1,2, 4,2, 4,5, 5,2, 5,5
2000 P=P+40:POKE P+2,205
2010 FOR Q=1 TO LN:P=P+41
2020 POKE P,205:POKE P+1,206:POKE P+2,205
2030 NEXT Q
2040 POKE P+41,205:RETURN
2100 P=P+40
2110 FOR Q=1 TO LN
2120 POKE P,234:POKE P+1,192:POKE P+2,244
2130 P=P+40:NEXT Q
2140 RETURN
2200 P=P+40:POKE P,206
2210 FOR Q=1 TO LN:P=P+39
2220 POKE P,206:POKE P+1,205:POKE P+2,206
2230 NEXT Q
2240 POKE P+39+2,206:RETURN
2300 P=P+81
2310 FOR Q=1 TO LN
2320 POKE P,223:POKE P+1,95:P=P+41
2330 NEXT Q
2340 POKE P,223:RETURN
2400 P=P+80
2410 FOR Q=1 TO LN
2420 POKE P,105:POKE P+1,233:P=P+39
2430 NEXT Q
2440 POKE P+1,233:RETURN
3000 ZQ=1:ZT=0
3010 GET IN$:IF IN$<>"" THEN PRINT "?~";:RETURN
3020 IF TI<ZT THEN 3010
3030 PRINT MID$( "?~",ZQ,1);":ZQ=3-ZQ:ZT=TI+15:GOTO 3010
60000 IN$=~":ZT=TI:ZC=2:ZD$=CHR$(20)
```

```

60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$( "~{+}" ,ZC,1) ; "{left}"; :ZC=3-ZC:
    ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}"; :
    GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}"; :
    GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
    GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
    CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
    TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$ :PRINT "{rvs-on}"
    TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
    MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$ :PRINT "{7°down cyn 9°space}
(C) ^1983 ^THE ^CODE ^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

BJACK

By John Melissa, Jr.



Have you wanted to improve your Blackjack skills? We have just the thing for you—a completely automatic blackjack dealer, trained in Las Vegas.

Here's a quick summary of the rules. Before the cards are dealt, you place your bet. You get a hand of two cards, both face up. The dealer then gets one card face up and one card face down. If you like, you can be *hit* and get another card. You may keep asking to be hit until the cards total 21 points or more. If you exceed 21, you *bust* and lose the hand. If you haven't busted, the dealer begins to play her hand. She must continue to take hits until her hand totals 17 or more or until she has five cards. If the dealer busts, you win. Otherwise, whoever has the higher hand wins. If you win, you get twice your bet back. If you lose, you lose your bet. If the hands are equal, then it's a standoff, and you get back your original bet.

Cards from 2 to 10 are worth their face value in points: 2, 3, 4, and so on. Jacks, queens, and kings are all worth 10 points. Aces are worth 1 or 11 points. The BJack program assumes you want the aces to be

worth 11 unless your hand would exceed 21. In that case, it counts aces as 1.

If your first two cards are an ace with a ten, jack, queen, or king, you have a *blackjack* and you win instantly. The payoff is one and a half times your bet. If the dealer has a blackjack, she wins.

If the dealer's visible card is an ace, you'll be offered *insurance* against her having a blackjack. Insurance is essentially a bet that the dealer has a blackjack. To buy insurance, you pay the dealer half the amount of your bet on that hand. If the dealer does have a blackjack, then you lose the hand but win the insurance bet. The net result is that you lose nothing. If she doesn't have a blackjack, you lose the insurance money.

If your first two cards total 10 or 11, you'll be given a chance to *double down*. If you do, the bet is doubled, and you get one final card. The round then proceeds as usual.

The game continues until you go broke or until you quit by betting 0.

```
1 PG$="^B^J^A^C^K":AU$="BY^JOHN^MELISSA, ^JR."
4 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/19/83 GF
90 GOTO 62000
100 DIM X(13),Y(13),D(52),B$(5),A$(14)
102 POKE VIC+32,12:POKE VIC+33,11
110 FOR I=1 TO 5:READ B$(I):NEXT I
115 DATA "{red Z cyn}","{blk A cyn}","{red S cyn}","{blk X cyn}",
" "
120 FOR I=1 TO 14:READ A$(I):NEXT I
125 DATA A,2,3,4,5,6,7,8,9,10,J,Q,K,?
130 BL$="{39°space}"
140 DN$="{home 21°down}"
145 PR$="{grn}HOW MUCH IN CHIPS?":GOTO 155
150 PR$="{grn}HOW MUCH MORE IN CHIPS?{shift-space}"
155 IN=500:GOSUB 9600:B2=IN:IF B2<.25 THEN 9100
```

```

160 B1=B1+B2
170 IF B>0 THEN 250
180 LB=INT(B1/10):PR$="FIRST BET?{shift-space}":GOTO 230
200 B1=B1+B:PRINT DNS;"{3°down}YOU ";MID$("LOSEWIN",-4*(B>0)+1,4);
205 PRINT "{shift-space}$";MID$(STR$(B),2);";YOU NOW HAVE $";
    MID$(STR$(B1),2);".{home}"
210 FOR J=1 TO 500:NEXT:IF B1<.01 THEN 9000
220 PR$="LAST BET WAS "+STR$(LB)+" ; NEW BET?{shift-space}"
230 PRINT DNS;"{2°up}"
240 IN=LB:GOSUB 9600:B=IN
245 IF IN$="" AND LB=0 THEN 230
250 IF B<.25 THEN 9100
260 IF B<=B1 THEN 300
270 PRINT "{clr down}YOU CAN'T BET";B;"{left}."
275 PRINT "{down}YOU HAVE ONLY";B1;"CHIPS."
280 PRINT "{down}GET MORE CHIPS!{down}":GOTO 150
300 LB=B:PRINT "{clr}BET ";B
310 F1=0:F2=0
320 FOR J=1 TO 52
330 LET D(J)=1
340 NEXT J
400 X(13)=14
410 Y(13)=5
500 REM
505 GOSUB 1000
510 X(1)=C1
515 Y(1)=C2
516 IF C1>10 THEN C1=10
517 M=C1
520 GOSUB 1000
530 X(7)=C1
535 Y(7)=C2
536 IF C1>10 THEN C1=10
537 N=C1
540 GOSUB 1000
550 X(2)=C1
560 Y(2)=C2
561 IF C1>10 THEN C1=10
563 M=M+C1
565 GOSUB 1000
570 X(8)=C1
575 Y(8)=C2
576 IF C1>10 THEN C1=10
577 N=N+C1
580 PRINT "{yel 12°*}"
600 J=7:P=1
605 GOSUB 2000
610 J=13:P=2

```

```

615 GOSUB 2000
620 PRINT "{6°down}":PRINT "{grn}DEALER"
670 PRINT "{yel 12°*}"
700 J=1:P=1:GOSUB 2000
720 J=2:P=2:GOSUB 2000
730 PRINT "{6°down}":PRINT "{grn}PLAYER"
735 M1=0
740 IF X(1)=1 OR X(2)=1 THEN M1=1
750 N1=0
755 GOSUB 2200:F1=1
760 IF X(7)=1 OR X(8)=1 THEN N1=1
770 I9=0:F1=1
775 REM
780 IF X(7)=1 THEN GOSUB 8000
790 IF N=11 AND N1=1 THEN 3000
800 IF I9=0 THEN 810
805 PRINT DN$;"{down}INSURANCE LOSES $" ;MID$(STR$(B/2),2);".":B1=B1-B/2
810 J=2
815 IF M=11 AND M1=1 THEN 5500
817 IF M=10 OR M=11 THEN GOSUB 7000:IF YN THEN 5000
820 IF M>20 THEN 5000
830 IF M=11 AND M1=1 THEN 5000
840 PRINT DN$
850 PRINT "{2°up grn}DO YOU WANT A HIT?{11°space 9°left}";
860 YN=1:GOSUB 9500
870 IF YN THEN GOSUB 4000:GOTO 820
880 GOTO 5000
890 GOTO 840
999 GOTO 9000
1000 C=INT(RND(1)*52+1)
1010 IF D(C)<>1 THEN 1000
1015 D(C)=0
1020 C1=INT(C/4)
1030 C2=C-C1*4
1032 C1=C1+1
1035 IF C2<1 THEN C2=C2+4:C1=C1-1
1099 RETURN
1900 PRINT "{home down}":REM FALLS INTO...
2000 REM
2020 PRINT TAB(P*7-7);
2030 PRINT "{cyn U 3°* I}";
2050 PRINT "{down left - down left - down left - down left - left
down - left down K 2°left * 2°left * 2°left * 2°left J left up
- left up - left up - left up - left up - }";
2120 PRINT A$(X(J));"{down left}";
2125 IF X(J)=10 THEN PRINT "{left}";
2130 PRINT B$(Y(J));

```

```
2150 PRINT "{right 2°down}";B$(Y(J));"{left down}";
2155 IF LEN(A$(X(J)))>1 THEN PRINT "{left}";
2160 PRINT A$(X(J));"{5°up}";
2190 IF F1=0 THEN RETURN
2200 REM
2210 M2=M
2220 IF M1=1 AND M<12 THEN M2=M+10
2230 PRINT "{home 18°down 7°right grn}=";M2
2300 REM
2310 IF F2=0 THEN RETURN
2320 N2=N
2330 IF N1=1 AND N<12 THEN N2=N+10
2340 PRINT "{home 9°down 6°right grn}~=";N2
2990 RETURN
3000 P=2:J=8
3040 GOSUB 1900
3042 IF I9=1 THEN 3100
3045 IF M=11 AND M1=1 THEN 5600
3050 PRINT DNS;"{down}DEALER HAS BLACKJACK"
3060 B=-B:GOTO 200
3100 N=0:IF M=11 AND M1=1 THEN 5620
3120 PRINT DNS;"{down}
    DEALER HAS BLACKJACK -- LOSS COVERED BY INSURANCE"
3130 GOTO 210
4000 J=J+1
4005 IF J>6 THEN 6000
4010 GOSUB 1000
4015 IF C1>10 THEN C1=10
4017 IF C1=1 THEN M1=1
4020 M=M+C1
4030 PRINT "{home 11°down}";
4040 P=J
4060 X(J)=C1
4070 Y(J)=C2
4090 GOSUB 2000
4400 RETURN
5000 J=8:P=2:F2=1
5020 GOSUB 1900
5030 IF M>21 THEN 5400
5090 J=J+1
5100 IF N>16 THEN 5600
5110 IF N<12 AND N1=1 AND N+10>16 THEN 5600
5120 GOSUB 1000
5130 X(J)=C1
5140 Y(J)=C2
5150 IF C1>10 THEN C1=10
5155 IF C1=1 THEN N1=1
5160 N=N+C1
```

```

5170 P=J-6
5180 GOSUB 1900
5186 FOR I=1 TO 400:NEXT
5190 GOTO 5090
5400 PRINT DN$
5410 PRINT "{11°space}*****~BUSTED~*****"
5420 B=-B:GOTO 200
5500 J=8:P=2
5530 GOSUB 1900
5560 PRINT DN$
5570 PRINT "{15°space}BLACKJACK!!"
5580 B=B*1.5:GOTO 200
5600 IF M<12 THEN IF M1=1 THEN M=M+10
5610 IF N<12 THEN IF N1=1 THEN N=N+10
5620 PRINT DN$
5625 IF N>21 THEN 5800
5630 IF M>N THEN 5700
5640 IF M<N THEN B=-B:GOTO 5700
5650 PRINT "{12°space}STAND-OFF"
5660 PRINT "{down}CHIP~BALANCE~REMAINS~AT~$";MID$(STR$(B1),2);
".{home}"
5699 GOTO 210
5700 PRINT DN$;"{down}DEALER~HAS";N;"{left}~YOU~HAVE";M;"{left}."
5710 GOTO 200
5800 PRINT DN$;"{down}***{shift-space}DEALER~BUSTED{shift-space}
***":GOTO 200
6000 PRINT DN$:PRINT "SIX~CARDS~LESS~THAN~21~IS~A~WINNER!"
6020 GOTO 200
7000 PRINT DN$
7010 PRINT "{2°up wht}DO~YOU~WANT~TO~DOUBLE?{rvs-off}~";
7020 GOSUB 9400:IF YN=0 THEN RETURN
7100 B=B*2
7110 PRINT "{home}BET~";B
7120 GOSUB 4000
7130 YN=1:RETURN
8000 PRINT DN$;"{up wht}WANT~INSURANCE?{rvs-off}~";
8010 GOSUB 9400:IF YN THEN I9=1
8020 RETURN
9000 PRINT DN$
9005 PRINT "YOU~ARE~BROKE.{25°space}"
9006 PRINT "{8°space 5°E 26°space}"
9010 PRINT "WOULD~YOU~LIKE~TO~PLAY~AGAIN?~";:GOSUB 60000
9020 IF LEFT$(IN$,1)="N" THEN 9100
9037 B=0
9040 PRINT "{clr}":GOTO 130
9100 PRINT "{clr}"
9120 END
9400 YN=0

```

```

9500 ZC=2:ZM=0
9510 GET Z$:IF Z$<>"" THEN 9545
9520 IF TI<ZM THEN 9510
9530 PRINT MID$("~{Q}",ZC,1);:{left}";:ZC=3-ZC
9540 ZM=TI+30:GOTO 9510
9545 IF Z$="Q" THEN 9100
9550 IF Z$<>"Y" AND Z$<>"N" AND Z$<>CR$ THEN 9520
9560 IF Z$<>CR$ THEN YN=Z$="Y"
9570 PRINT MID$("YN",2+(YN<>0),1)
9580 RETURN
9600 PRINT BL$:PRINT "{up grn}";
9605 CR$="":PRINT PR$;:GOSUB 60000:IF IN$="" THEN Z=IN:
    PRINT "{left}";IN:GOTO 9640
9606 IF IN$="Q" THEN 9100
9610 PRINT:FOR Z=1 TO LEN(IN$):Z$=MID$(IN$,Z,1)
9620 IF Z$<"0" OR Z$>"9" THEN 9650
9630 NEXT Z:Z=VAL(IN$)
9635 IF Z/10<>INT(Z/10) THEN PRINT "{down wht}
    A^MULTIPLE^OF^TEN,^PLEASE. ";:GOTO 9660
9640 CR$=CHR$(13):IN=Z:RETURN
9650 PRINT "{down wht}NO^";Z$;'"S^PLEASE.";
9660 PRINT MID$(BL$,POS(0)+1):FOR Z=1 TO 2000:NEXT Z
9670 PRINT "{up}";BL$:PRINT "3°up";
9680 GOTO 9600
60000 IN$=~":ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);:{left}";:ZC=3-ZC:
    ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
    GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
    GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
    GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
    CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
    TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
    TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$

```

```
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
    MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$ : PRINT "{7°down cyn 9°space}
(C) ^1983 ^THE ^CODE ^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("pur cyn",K+1)LEFT$(MS$,
Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```



FIRE (Uses joystick)

By Brian Sawyer

There is a fire in a downtown high-rise building, and you have three minutes to put it out. Your fire-fighting helicopter carries one thousand gallons of water, which is enough to put out the fire if you can get to it quickly. The longer the fire burns, the more it spreads, and the more water it takes to put it out.

If you run out of water, you'll have to fly back to the

water supply station at the right of the screen and refill your tanks. When you want to refill, you must land the helicopter by using the joystick to move down over the refilling station. Move the joystick handle left or right to move the helicopter. The fire will continue to rage while you are at the station, so avoid refills if you can. Press the joystick button to drop water on the fire.

```
0 POKE 56,44:CLR:PG$="^F^I^R^E^":AU$="BY^BRIAN^SAWYER":JF=1
1 REM FIRE -- BY BRIAN SAWYER
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/11/83 GF
90 GOTO 62000
100 DIM JD(15):FOR I=0 TO 15:READ JD(I):NEXT
101 DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,8,5
105 JS=56320:DEF FNJ(X)=JD(PEEK(JS+X) AND 15):
    DEF FNB(X)=(PEEK(JS+X) AND 16)=0
110 A=53248:Z=0:P=RND(-TI):READ L:ML=50*1024
115 SID=54272:FOR I=0 TO 28:POKE SID+I,0:NEXT:POKE SID+24,15:
    POKE SID+6,129
120 POKE SID+20,129:POKE 2042,192:POKE A+21,12:POKE A+29,4:
    POKE A+32,0
125 POKE A+33,0:POKE A+41,14
130 POKE 2043,193:POKE A+42,3
135 C1$="{pur}":C2$="{yel}":C3$="{blu}"
140 VL=SID+24:CL=55296
145 FOR I=0 TO L-1:READ X:POKE ML+I,X:NEXT I
150 REM READ IN SPRITE DATA
155 FOR I=0 TO 254:READ X:POKE 12288+I,X:NEXT
160 REM SYS AT 980
180 P=CRT+WD*13+4:E=29:V=3:TI$="000000":WT=1000:BP=P:CP=P:
    WL$="{3°left up rvs-on}^":I=4
190 FY=13:FX=4:PRINT "{clr 6°down}":POKE 53248+4,E*8+15:
    POKE A+21,12
200 PRINT "{27°space blu 2°@ grn 3°I blu 2°@}^"
210 PRINT C1$+"{right rvs-on M 8°shift-space C rvs-off}
    ^~{rvs-on}"+C2$+"{M 9°shift-space C rvs-off}~"+C3$+"{rvs-on
    M shift-space X shift-space Z 2°shift-space C}"
```

```

220 PRINT Cl$+"{right rvs-on 2°M 8°shift-space C rvs-off}
  ~~{rvs-on}"+C2$+"^M 9°shift-space C rvs-off}~"+C3$+"{rvs-on}
  ^M 6°shift-space C}"
230 PRINT Cl$+"{right rvs-on 2°M O 8°Y rvs-off}~~{rvs-on}"+C2$+"{M
  shift-space O 8°Y P rvs-off}~"+C3$+"{rvs-on _}~{O 5°Y P}"
240 PRINT Cl$+"{right rvs-on 2°M O 8°Y rvs-off}~~{rvs-on}"+C2$+"{M
  2°G shift-space OG shift-space OGMPM rvs-off}~"+C3$+"{rvs-on}
  ^{-G+}~{+}~{+M}"
250 PRINT Cl$+"{right rvs-on 2°M O 8°Y rvs-off}
  ~~{rvs-on}"+C2$+"{2°shift-space G shift-space Y 2°shift-space
  Y 2°shift-space YM rvs-off}~"+C3$+"{rvs-on}~~{G 5°space M}"
251 PRINT Cl$+"{right rvs-on 2°M O 8°Y rvs-off}
  ~~{rvs-on}"+C2$+"{MMG shift-space OG shift-space OGMPM rvs-off}
  ~"+C3$+"{rvs-on _}~{G+}~{+}~{+M}"
252 PRINT Cl$+"{right rvs-on 2°M O 8°Y rvs-off}
  ~~{rvs-on}"+C2$+"{MG shift-space Y 2°shift-space Y
  2°shift-space YM rvs-off}~"+C3$+"{rvs-on}~~{G 5°space M}"
260 PRINT Cl$+"{right rvs-on 2°M O 8°Y rvs-off}
  ~~{rvs-on}"+C2$+"{MMG shift-space OG shift-space OGMPM rvs-off}
  ~"+C3$+"{rvs-on _}~{G+}~{+}~{+M}"
270 PRINT Cl$+"{right rvs-on 2°M O 8°Y rvs-off}
  ~~{rvs-on}"+C2$+"{2°M G shift-space Y 2°shift-space Y
  2°shift-space YM rvs-off}~"+C3$+"{rvs-on}~{-G 5°space M}"
280 PRINT Cl$+"{right rvs-on 2°M O 8°Y rvs-off}~~{rvs-on}"+C2$+"{M
  2°G}~{OG}~{OGMPM rvs-off}~"+C3$+"{rvs-on}~~{G+}~{+}~{+M}"
290 PRINT Cl$+"{right rvs-on 2°M O 8°Y rvs-off}
  ~~{rvs-on}"+C2$+"~~{G}~{Y}~{Y}~{YM rvs-off}~"+C3$+"{rvs-on _}
  ~{G 5°space M}"
300 PRINT Cl$+"{right rvs-on 2°M O 8°Y rvs-off}
  ~~{rvs-on}"+C2$+"{MMG}~{OG}~{OGMPM rvs-off}~"+C3$+"{rvs-on}
  ~{-G+}~{+}~{+M}"
310 PRINT Cl$+"~{* rvs-on MO 8°Y rvs-off M}~"+C2$+"{* rvs-on MG}
  ~{Y}~{Y}~{YM rvs-off M}~"+C3$+"{* rvs-on}~{G}~{@}~{M}"
320 PRINT Cl$+"~~{* rvs-on O 3°Y 2°O 3°Y rvs-off}~~"+C2$+"{* rvs-on
  L 4°@ OL 2°@ @ rvs-off shift-space}~"+C3$+"{* rvs-on
  L@@+L@@}"
330 PRINT "{3°right 34°I}"
340 PRINT "~~-----{4°space}-----~~-----"
345 FOR NY=11*WD TO 20*WD STEP WD*2:FOR NX=18 TO 26 STEP 3:
  POKE CL+NY+NX,6
350 NEXT:NEXT
360 IF TI>10800 THEN 1060
370 IF WT=0 THEN WK=WK+1:POKE A+41,14:IF WK/2=INT(WK/2) THEN
  POKE A+41,8
390 IF FX=10 THEN FX=16:P=P+6:GOTO 510
400 IF FX=24 THEN FX=16:P=P-6:GOTO 510
420 REM IFINT(RND(10)*2)=0THEN510

```

```

430 CY=0:IF RND(1)>.5 THEN CY=-1:GOTO 450
435 GET IN$:IF IN$="Q" THEN POKE VIC+21,0:POKE VIC+5,0:
    PRINT "{clr}":END
440 IF RND(1)>.5 THEN CY=1
450 CX=0:IF RND(1)>.5 THEN CX=1:GOTO 460
455 IF RND(1)>.5 THEN CX=-1
460 G=CX+CY*WD:IF PEEK(P+G)=32 OR PEEK(P+G+1)=32 OR FX+CX<3 THEN
    510
465 IF (PEEK(P+G)=160 OR PEEK(P+G+1)=160) AND RND(1)<.7 THEN 510
500 P=P+G:FX=FX+CX
505 POKE P,233:POKE P-CRT+CL,2:POKE P+1,223:POKE P+1-CRT+CL,
    8+INT(RND(10)*2)*5
510 GOSUB 970:REM : GETI$:I=VAL(I$):REM :IFI1THENI=I1
515 REM IF I1=0ANDI=5THENI=0
520 IF I=6 AND E<30 THEN E=E+1
530 IF I=4 AND E>3 THEN E=E-1
540 IF I=2 AND E=30 THEN V=V+1
565 POKE 53248+4,E*8+15:POKE 53248+5,V*8+60
570 PRINT:IF NOT (E=30 AND V=4) THEN 700
590 POKE SID+24,15:POKE A+41,14:IF WT>=1000 THEN 630
599 POKE SID+19,64:POKE SID+18,17
600 FOR WT=WT TO 1000 STEP 100:POKE SID+14,WT/20:
    POKE SID+15,WT/20+20:GOSUB 970
620 FOR N=1 TO 100:NEXT:NEXT:POKE SID+18,0:POKE SID+19,0:
    POKE SID+14,0:WT=1000
630 D=4:V=V-1:I=4
700 GOSUB 970
710 I=FNJ(0)
720 IF FNB(0)=0 THEN 360
750 IF WT>0 AND E<>30 THEN 810
800 POKE VL,0:GOTO 360
810 WT=WT-50:POKE SID+5,64:POKE SID+4,17:POKE SID+24,15
860 POKE A+6,E*8+18:FOR W=V*8+68 TO 8*26 STEP 8:POKE 2043,193:
    POKE SID,235-W
870 POKE SID+1,230-W:WS=(CRT+E+INT(((W-40)/8))*WD)
880 IF PEEK(WS)=223 OR PEEK(WS)=233 THEN POKE WS,160:
    POKE WS-CRT+CL,11:GOTO 900
890 POKE 2043,194:POKE A+7,W:NEXT:POKE 2043,195:FOR I=1 TO 200:NEXT
895 POKE A+6,0:POKE A+7,0:POKE SID+4,0:POKE SID+5,0:POKE SID,0:
    GOTO 360
900 POKE 2043,195
910 WS=WS+WD:IF PEEK(WS)=223 OR PEEK(WS)=233 THEN POKE WS,160:
    POKE WS-CRT+CL,11
920 FOR N=15 TO 0 STEP -1:POKE SID+24,N:POKE SID+4,129:
    POKE SID+5,15
930 POKE SID+1,40:POKE SID,200:NEXT
935 POKE SID+4,0:POKE SID+3,0:POKE SID+24,15
940 POKE A+7,0:POKE A+6,0:SYS ML

```

```

950 IF PEEK(10)=0 THEN 1000
960 GOTO 360
970 PRINT "{pur home down}";TAB(6);"{rvs-off}TIME_";MID$(TI$,4,1);
":";
975 PRINT RIGHT$(TI$,2);SPC(8);"WATER:";WT;"{left}_"
980 SYS ML:RETURN
990 REM
1000 TM$=TI$:POKE A+21,0
1010 GET D$:IF D$<>"" THEN 1010
1020 FOR N=1 TO 3000:NEXT
1030 PRINT "{clr 3°down}^YOU^PUT^THE^FIRE^OUT^IN^";MID$(TM$,4,1);
":";RIGHT$(TM$,2);"!"
1040 XT=XT+1:GOTO 1070
1050 FOR N=1 TO 3000:NEXT
1060 POKE A+21,0:PRINT "{clr 2°down}
^YOU^FAILED^TO^PUT^OUT^THE^FIRE!"
1070 XN=XN+1:PRINT "{down 2°right}YOU^HAVE^PUT^OUT" XT "OUT^OF" XN
"FIRES."
1080 GOSUB 9100:IF YN THEN 180
1090 PRINT "{clr}":END
1120 DATA 54,160,0,132,10,160,1,169,0,133,251,168,162,4,134,252,177,
251
1130 DATA 201,223,208,6,132,10,169,233,208,16
1140 DATA 201,233,208,4,132,10,169,223,208,6
1160 DATA 201,205,208,4,169,205,145,251
1170 DATA 200,208,223,230,252,202,208,218,96
9100 NO=0:PRINT "{wht 3°down}^PLAY^AGAIN?^{\u0016cyn}YES^{\u0016up}":
POKE VIC+5,0
9110 TM=0:SW=1
9115 PRINT "{wht}^{\u0016cyn}YES^{\u0016up}"
9120 IF TI<TM THEN 9150
9130 PRINT TAB(14+NO*4);MID$("{\wht \cyn}",SW,1);
MID$("YESNO{\shift-space}",3*NO+1,3);"\u0016up"
9140 SW=3-SW:TM=TI+15
9150 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
9160 IF JB AND JD=15 THEN YN=(NO=0):RETURN
9170 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 9110
9180 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 9110
9190 GOTO 9120
50000 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,63
50010 DATA 128,0,4,0,0,4,0,0,14,0,0,63,128,6,79,224
50020 DATA 11,143,255,253,159,255,253,255,224,11,127,192,6,9,0,0
50030 DATA 9,0,0,63,192,0,0,0,0,0,0,0,0,0,0,0,0,0,0
50040 DATA 0,0,0,0,0,0,48,0,0,48,0,0,48,0,0
50050 DATA 48,0,0,50,0,0,51,0,2,51,0,6,35,0,6,3
50060 DATA 0,6,11,0,6,43,0,6,107,0,2,98,0,0,96,0
50070 DATA 0,96,0,2,96,0,2,32,0,2,0,0,0,0,0,0,0
50090 DATA 0,0,0,0,0,0,32,0,0,96,0,2,96,0,6

```

```

50100 DATA 98,0,6,6,0,6,102,0,0,102,0,12,100,0,12,33
50110 DATA 0,12,35,0,4,3,0,0,130,0,0,192,0,0,200,0
50120 DATA 0,204,0,0,76,0,0,12,0,0,4,0,0,0,0,0,0
50140 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,96,6
50150 DATA 0,192,7,3,129,3,199,6,1,195,12,1,129,56,252,0
50160 DATA 112,62,0,64,0,1,0,14,0,192,28,68,112,48,198,24
50170 DATA 97,131,12,3,3,0,0,1,128,0,0,128,0,0,0,255,255,0,255
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);:{left}";:ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
(C) ^1983 ^THE ^CODE ^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```



ZAP (Uses joystick)

By Tom Marazita and Brian Sawyer

Your goal in this game is simple: try to "zap" all the targets that pop on and off the screen. When the game begins, a snake-like train of dots (your zapper) starts moving up the right side of the screen. You can control the zapper easily with the joystick, but there are a few complications: you must not run into a wall or turn back across your zapper.

Each time you zap a target, your score goes up, the zapper grows longer, and the remaining targets increase in value. Eventually, your zapper will get so long that you'll have trouble keeping out of your own way. The game is over when you crash into your zapper six times.

```
0 PG$="{shift-space}Z~A~P~":  
AU$="BY~TOM~MARAZ~ITA~AND~BRIAN~SAWYER":JF=1  
1 REM ZAP -- BY TOM MARAZ ITA  
2 :  
5 REM COPYRIGHT (C) 1983 THE CODE WORKS  
6 REM BOX 6905, SANTA BARBARA, CA 93160  
7 :  
10 REM AS OF 9/1/83 GF  
90 GOTO 62000  
100 MX=255:DIM WM(MX),A(MX):BS=0:QI=256  
101 DIM JD(15):FOR I=0 TO 15:READ JD(I):NEXT  
102 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND  
16)=0  
103 GOSUB 7000:GOSUB 4000:DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,8,5  
110 EL=18:HP=0:TP=0:WI=2:LM=MX-WI:AL=0:CA=0:H=0:TC=0:SC=0:K=0:Q=0:  
KN=0  
111 GOSUB 1000  
120 POKE 51147,(5-G)*3  
125 PRINT "{clr}":SYS 51165  
130 POKE 251,190:POKE 252,7:WM(HP)=PO:HP=HP+1:AL=AL+1:S=1:LN=10:  
K=0:NT=0  
140 POKE 1001,216:POKE 51061,10:FOR N=50000 TO 50555:POKE N,0:NEXT  
160 FOR X=WD+1 TO WD*2-2:POKE CRT+X,100:POKE CM+X,1:NEXT  
170 FOR X=WD*2 TO WD*23 STEP WD:POKE CRT+X,103:POKE CM+X,1:NEXT  
180 FOR X=WD*24+1 TO WD*25-2:POKE CRT+X,99:POKE CM+X,1:NEXT  
190 FOR X=WD*24-1 TO WD*3-1 STEP -WD:POKE CRT+X,101:POKE CM+X,1:  
NEXT  
191 POKE CRT+WD,96:POKE CRT+WD+39,96:POKE CRT+WD*24,96:  
POKE CRT+WD*24+39,96  
192 PRINT "{home}" TAB(31) "{wht}LEVEL=" G  
195 GOSUB 900  
211 GOTO 380  
230 GET A$:IF A$="Q" THEN 650
```

```

326 SYS 51003:FOR DL=1 TO (5-G)*30:NEXT
327 Z=PEEK(999):IF Z=42 THEN GOSUB 550:GOTO 350
328 IF Z<>32 THEN 590
329 IF NT>0 AND RND(1)<.93 THEN 326
350 IF K>4 AND RND(1)<G/10 THEN 230
360 IF K>9 AND RND(1)<.85 THEN 405
380 K=K+1:KN=KN+1:NT=NT+1
390 A(K)=RND(1)*904+1104:IF PEEK(A(K))<>32 THEN 390
395 POKE A(K),42:GOTO 230
400 IF K=0 THEN 380
405 Z=INT(RND(1)*K)+1
410 REM IF PEEK(A(Z))<>42 THEN 440
415 IF PEEK(A(Z))<>42 THEN A(Z)=A(K):K=K-1:GOTO 440
420 IF RND(1)>NT/9 THEN 230
430 NT=NT-1:POKE A(Z),32
440 A(Z)=A(K):K=K-1:GOTO 230
550 POKE SID+18,33:NT=NT-1
560 SC=SC+INT(LN/1.6):Q=Q+1
565 S=S-1:POKE (PEEK(251)+PEEK(252)*256),81:LN=LN+2:IF LN>127 THEN
LN=127
567 POKE 51061,LN:IF EL<LM THEN EL=EL+WI
570 GOSUB 900:POKE SID+18,16
580 RETURN
590 CA=CA+1
600 IF CS>L THEN L=CS
610 CS=0
615 EL=18:HP=0:AL=0
616 TP=0
620 CV=0:POKE SID+4,129:POKE SID+11,0
625 PO=PEEK(251)+PEEK(252)*256
635 FOR T=1 TO 8:FOR X=1 TO 100:NEXT:POKE SID+4,128:POKE PO,209:
FOR X1=1 TO 100
636 NEXT:POKE PO,81:NEXT:POKE SID+24,0:GOSUB 1000
638 IF CA=6 THEN 650
640 GOTO 120
650 IF SC>BS THEN BS=SC
660 PRINT "{clr 7°down}YOUR SCORE IS" SC
665 PRINT "{down}YOU HIT";INT((Q/KN)*100);"{left}% OF THE TARGETS"
670 PRINT "{2°down}HIGHEST SCORE SO FAR:";BS:PRINT
680 GOSUB 3500
682 IF YN THEN 110
700 PRINT:END
900 PRINT "{home right}SCORE=" SC;TAB(13);"CRASHES LEFT=" 5-CA
910 RETURN
1000 FOR I=0 TO 28:POKE SID+I,0:NEXT I
1100 REM CRASH
1110 POKE SID+6,250
1120 POKE SID+1,2

```

```

1200 REM BELL
1210 POKE SID+15,64
1220 POKE SID+20,250
1300 REM TURN
1310 POKE SID+12,80
1320 POKE SID+13,240
1330 POKE SID+8,12
1400 POKE SID+24,15
1500 RETURN
3200 T=PEEK(VIC+17):POKE VIC+17,0:T=T/1
3210 POKE VIC+33,7:PRINT "{clr}"
3220 POKE VIC+33,0:POKE VIC+17,T
3230 RETURN
3500 NO=0
3510 TM=0:SW=1:PRINT "{wht}TRY AGAIN?{shift-space cyn}YES {up}"
3520 IF TI<TM THEN 3550
3530 PRINT TAB(11+NO*4);MID$("{wht cyn}",SW,1);
    MID$("YESNO{shift-space}",3*NO+1,3);"{up}"
3540 SW=3-SW:TM=TI+15
3550 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
3560 IF JB AND JD=15 THEN 3600
3570 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 3510
3580 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 3510
3590 GOTO 3520
3600 PRINT TAB(11+NO*4);"{wht}";MID$("YESNO{shift-space}",3*NO+1,3);
    "{up}":YN=(NO=0):RETURN
4000 PRINT "{wht}SETTING UP...{left}"
4010 FOR N=51000 TO 52000:READ A:IF A=999 THEN RETURN
4020 POKE N,A:NEXT
4090 RETURN
7000 PRINT "{clr wht 2°down}PICK A SKILL LEVEL:{down cyn}":
7010 FOR I=1 TO 5:PRINT I;:NEXT I:PRINT:G=1:T=1:TM=0
7020 X=G:IF TI>TM THEN PRINT TAB(3*G-3);MID$("{wht cyn}",T,1);
    "{up}";G:T=3-T:TM=TI+10
7030 IF FNJ(0)=4 THEN IF G>1 THEN G=G-1
7040 IF FNJ(0)=6 THEN IF G<5 THEN G=G+1
7050 IF X<>G THEN PRINT TAB(3*X-3);"cyn up";X:T=1:TM=0
7060 IF FNB(0)=0 THEN 7020
7070 PRINT "{clr}":RETURN
50000 DATA 32,202,199,32,155,199,173,233,3,201,128,16,14,24,101,251,
    133
50010 DATA 251,165,252,105,0,133,252,24,144,9,24,101,251,133,251,176,
    2
50020 DATA 198,252,160,0,177,251,141,231,3,169,81,145,251,32,137,199,
    165
50030 DATA 251,141,80,195,165,252,141,124,196,162,10,189,80,195,133,
    253,189

```

```

50040 DATA 124,196,133,254,240,6,169,32,160,0,145,253,96,162,250,189,
79
50050 DATA 195,157,80,195,189,123,196,157,124,196,202,208,241,96,162,
0,173
50060 DATA 0,220,74,176,2,162,1,74,176,2,162,2,142,234,3,74,176
50070 DATA 7,168,138,24,105,3,170,152,74,176,5,138,24,105,6,170,224
50080 DATA 0,240,6,189,212,199,141,233,3,96,169,0,24,101,162,197,162
50090 DATA 208,252,96,1,216,40,255,215,39,1,217,41,169,216,133,254,
169
50100 DATA 80,133,253,162,22,160,38,152,41,3,142,237,3,170,189,10,200
50110 DATA 145,253,174,237,3,136,208,238,165,253,24,105,40,133,253,
144,2
50120 DATA 230,254,202,208,222,96,7,7,7,7,999
60000 IN$=".":ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);:ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
(C) ^1983 ^THE ^CODE ^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

EVEREST

(Uses joystick)

By Brian Sawyer



The object of EVEREST is to climb to the top of Mt. Everest before Mother Nature kills you. As you start the climb, you'll see a white and gray mountain. The rocks (shown as darker patches) are easier to climb and should be followed as much as possible. You climb by using the joystick to make your way up the face of the peak. Now and then an avalanche will come roaring down the mountain, or a fierce snowstorm will threaten your climb. Both storms and avalanches will knock you down the mountain a little, but avalanches tend to knock the snow away and leave more rocks exposed. Snowstorms, however, just add to your problems by drop-

ping more snow in your path.

Since you are a wealthy mountain climber, you can have extra food airlifted to you as many as three times during your climb. To call in a plane, press the joystick button. An airplane will fly across the top of the screen and drop supplies near you. When you climb over and touch the supplies, your energy level is increased, and you can resume your climb. Remember, though, that it will take some energy to climb over to the food.

The game ends when you reach the top or when you finally succumb to exhaustion while clinging to those windswept rocks. Have fun.

```
0 PG$="^E^V^E^R^E^S^T^":AU$="BY^BRIAN^SAWYER":JF=1
1 REM EVEREST -- BY BRIAN SAWYER
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/11/83 GF
90 GOTO 62000
100 CL=55296:V=53248:DIM JD(15):FOR I=0 TO 15:READ JD(I):NEXT
105 DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,8,5
110 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
16)=0
115 DIM V(10):FOR N=0 TO 9 STEP 2:V(N)=204:V(N+1)=205:NEXT:
P(0)=200:P(1)=203
120 POKE 53280,3:POKE 53281,0:POKE V+40,7:FOR N=1 TO 6:READ A(N):
NEXT
130 DATA 0,11,12,15,1,1
131 FOR BS=200*64 TO 208*64 STEP 64
132 FOR N=0 TO 62:POKE BS+N,0:NEXT
133 READ A:FOR N=0 TO A-1:READ B:POKE N+BS,B:NEXT:NEXT
135 DIM C(15):C(0)=1:C(11)=2:C(12)=3:C(15)=4:C(1)=5
136 DIM K(5):K(1)=1:K(2)=1:K(3)=1:K(4)=11:K(5)=11
140 POKE 2042,205:POKE 2040,201:POKE 2041,200:POKE 2043,207:
POKE 53277,5
150 SR=100:POKE 2044,208:GOSUB 1000:GOSUB 2238
165 X=113:Y=238:DR=3:POKE V+21,31:GOSUB 2239
```

```

167 VL=54296:NO=54272:WV=54276:DE=54277:POKE VL,15:SU=54278:
    POKE DE+7,129
168 POKE NO+7,0:POKE NO+8,0:POKE SU+7,255:POKE WV,129:POKE WV+7,129
170 GOSUB 2400:GOSUB 2000:IF SR>0 AND Y>73 THEN 170
175 IF Y<74 THEN 4000
180 GOTO 4100
1000 REM DRAW MTN
1010 PRINT "{clr}":GOSUB 1300:GOSUB 1600:XN=1:P1=28-4*WD
1020 FOR Y=1 TO 21:FOR X=1 TO XN:P=Y*WD+X-P1:N=INT(RND(1)*5)+1
1025 IF RND(1)>.7 THEN N=5-INT((Y-1)/4)
1030 POKE P+CRT,160:POKE P+CL,A(N):NEXT X
1035 XN=XN+2:P1=P1+1:IF Y>13 THEN P1=P1-1:XN=XN-1
1040 NEXT Y:POKE CL+13+4*WD,1
1050 PRINT "home 22°down 31°right";
1060 PRINT "~{down left}~{down 2°left}~~{home}"
1090 RETURN
1095 POKE P,4:IF NOT FNB(0) THEN 4230
1300 PRINT "{grn 5°down 36°space NM}"
1305 PRINT "{31°space NM}~{N}~{M}"
1310 PRINT "{28°space NMN}~{MNMNM}"
1315 PRINT "{27°space N}~{M}~{N 4°space M}"
1320 PRINT "{26°space N 4°space MN}"
1325 PRINT "{25°space NMNMNMNM}"
1330 PRINT "{24°space N 8°space M}"
1340 PRINT "{34°space M}"
1345 PRINT "{35°space M}"
1350 PRINT "{36°space M}"
1355 PRINT "{37°space M}"
1360 PRINT "{38°space M}"
1370 RETURN
1400 REM
1405 PRINT "{grn 31°space NM}"
1410 PRINT "{30°space N}~{M}"
1415 PRINT "{29°space NMNMNM}~{NM}"
1425 PRINT "{28°space N 6°space N}"
1430 PRINT "{27°space N}~{NMNMN}"
1435 PRINT "{29°space N}~{N}~{M}"
1440 PRINT "{28°space N}~{N 4°space M}"
1445 PRINT "{29°space N 6°space M}"
1450 PRINT "{37°space M}"
1455 PRINT "{38°space M}"
1460 PRINT "{26°space}"
1500 RETURN
1600 PRINT "home 16°down rvs-on wht £ up £ up £ up £ up £ up £ up
    £ up £ up £ up £ up £ up £ up £}";
1610 PRINT "{right * down * down * down * down * down * down * down
    * down * down * down * rvs-on down * down * down * down * down
    * down * down * home}":RETURN

```

```
2000 REM BLIZZARD
2005 BY=INT(RND(10)*135)+95:BX=216-BY:D=3:IF RND(1)>.5 THEN D=-3:
    BX=BY+26
2010 E=RND(1)*5-1:B=CL+BX/8+INT(BY/8)*WD-241
2110 BY=BY+E:BX=BX+D:POKE V+39,6+8*(BX AND 1)
2115 POKE B,1
2120 POKE 2040,201+RND(1)*2:POKE V,BX AND 255:POKE V+1,BY:GOSUB 2200
2125 IF NOT (BY>60 AND BY<230 AND BX>24 AND BX<255) THEN 2140
2130 IF SR AND Y>72 AND RND(1)>.09 THEN GOTO 2010
2140 POKE V+1,0:RETURN
2200 REM JOYSTICK
2201 GET Q$
2202 IF Q$="Q" THEN GOSUB 4300:GOTO 4210
2205 POKE V+2,X:POKE V+3,Y:Z=PEEK(53278):IF Z AND 2 THEN GOSUB 2700
2206 IF NOT PEEK(53279) AND 2 THEN GOSUB 2800
2207 IF FNB(0) THEN GOSUB 2600
2210 N=FNJ(0):IF N=5 AND SR>.9 THEN RETURN
2215 POKE NO+8,SR+3:IF N=6 AND X<247 THEN X=X+8:GOTO 2234
2220 IF N=4 AND X>9 THEN X=X-8:GOTO 2234
2230 IF N=8 THEN Y=Y-1:GOTO 2234
2232 IF N=2 AND Y<238 THEN Y=Y+2
2234 A=C(PEEK(CL+X/8+INT((Y+3)/8)*WD-241) AND 15):SR=SR-A*.45
2237 POKE 2041,P(R AND 1):R=R+1
2238 IF SR<1 THEN SR=0
2239 PRINT "home down red" TAB(5) "STRENGTH:" INT(SR) "left"
2240 PRINT "home down red" TAB(22) "PLANES LEFT:" INT(DR) "left"
    ~:POKE NO+8,0:RETURN
2400 REM MAKE AVAL
2410 AY=INT(RND(10)*100):AX=114+AY*SGN(RND(1)-.5):AY=AY+90
2420 POKE V+4,AX:POKE V+5,AY
2425 POKE NO,255:POKE NO+1,37:POKE WV,129:POKE DE,4+8+128+64:
    POKE SU,1
2430 KL=CL+AX/8+INT(AY/8)*WD-241:AZ=AY+RND(1)*(224-AY)
2510 FOR AH=AY TO AZ STEP 4:POKE V+5,AH:POKE 2042,V(AH AND 7)
2520 POKE KL,A(RND(1)*3):POKE KL+1,A(RND(1)*3):IF AH AND 4 THEN
    KL=KL+WD
2530 GOSUB 2200:IF KL>CL+40*24 THEN STOP
2535 IF Y<72 OR SR=0 THEN RETURN
2540 NEXT:POKE V+5,0:POKE V+4,0:POKE WV,0:POKE SU,0:RETURN
2600 REM INIT PLANE DROP
2601 IF DR=0 THEN RETURN
2605 DR=DR-1:POKE V+9,0:A=INT(RND(10)*70)+100:POKE WV+7,129
2610 B=65:POKE V+7,50:FOR N=255 TO 0 STEP -1:POKE V+6,N:
    POKE NO+8,N/10
2615 IF N=A THEN POKE V+8,A:GOTO 2620
2616 NEXT
2620 POKE WV+7,17:FOR N=A TO 0 STEP -1:POKE V+9,B:B=B+.5:POKE V+6,N:
    POKE NO+8,N
```

```

2622 NEXT
2625 B=B+1:POKE V+9,B
2630 IF (B<Y-22) OR (PEEK(53279) AND 16=0) THEN 2625
2640 POKE WV+7,129:POKE 2044,208:GOSUB 2240:RETURN
2700 REM
2704 IF (Z AND 16)=0 OR (PEEK(V+9)=0) THEN 2708
2705 IF Z<>18 THEN RETURN
2706 POKE V+9,0:SR=SR+50:IF SR>100 THEN SR=100
2707 GOSUB 2239:C=PEEK(53278):RETURN
2708 IF TW+130>TI THEN RETURN
2710 IF Y<210 THEN FOR Y=Y TO Y+12:POKE NO+8,255-Y:POKE V+2,X:
    POKE V+3,Y:NEXT
2711 POKE NO+8,0:TW=TI:RETURN
2800 FOR Z=Y TO 255:POKE V+3,Z:IF PEEK(53279) AND 2 THEN Y=Z:RETURN
2810 NEXT:RETURN
3200 REM DELAY
3205 STOP:FOR H=1 TO 40:NEXT:RETURN
4000 REM TOP OF MOUNTAIN
4005 FOR N=1 TO 1000:NEXT
4010 GOSUB 4300:PRINT "{clr 2°down 6°space}"
    YOU^CONQUERED^MT.^EVEREST!:WN=WN+1
4020 GOTO 4200
4100 REM YOU LOSE
4110 GOSUB 4300:PRINT "{clr 2°down 11°space}OUT_OF_ENERGY!"
4120 PRINT "{down}^YOU^FAILED^TO^CONQUER^MT.^EVEREST!"
4200 PRINT "{down}":IF AS THEN PRINT "YOU'VE^SUCCEEDED^IN" WN
    "OUT_OF" AS "CLIMBS"
4205 GOSUB 9100:IF YN THEN 140
4210 POKE VIC+21,0:PRINT "{clr}":END
4300 REM TURN EVERYTHING OFF
4310 POKE V+9,0:POKE VL,0:POKE V+1,0:POKE V+3,0:POKE V+5,0:RETURN
9100 NO=0:PRINT "{wht 3°down}^PLAY AGAIN?^{\u00c3\u00b3n}YES^{\u00c3\u00b3p}"
9110 TM=0:SW=1
9115 PRINT "{wht}^PLAY AGAIN?^{\u00c3\u00b3n}YES^{\u00c3\u00b3p}"
9120 IF TI<TM THEN 9150
9130 PRINT TAB(14+NO*4);MID$("{wht \u00c3\u00b3n}",SW,1);
    MID$("YESNO{\shift-space}",3*NO+1,3);"\u00c3\u00b3p"
9140 SW=3-SW:TM=TI+15
9150 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
9160 IF JB AND JD=15 THEN YN=(NO=0):RETURN
9170 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 9110
9180 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 9110
9190 GOTO 9120
50000 DATA 39,0,0,57,0,0,59,0,0,51,0,3,255,0,3,127,0,3,120,0,3,120,0,
    2,124,0,0
50001 DATA 110,0,0,102,0,0,102,0,0,102,0,0,96
50002 DATA 27,1,8,128,4,65,16,34,148,164,136,68,82,74,41,73,146,68,
    160,4,146

```

```

50003 DATA 138,81,41,80,1,16,128
50004 DATA 30,0,198,64,25,49,140,162,16,214,154,99,144,9,210,99,196,
89,156,10
50005 DATA 162,36,177,3,103,50,201,12,0,204,192
50006 DATA 36,0,0,60,0,3,60,0,1,152,0,1,254,0,0,255,0,0,123,0,0,251,
0,1,249,0,1
50007 DATA 152,0,3,152,0,0,24,0,0,24
50008 DATA 50,37,155,0,109,155,0,109,155,0,109,155,0,108,155,0,104,
59,0,64,2,0
50009 DATA 13,162,0,19,88,0,219,210,0,134,205,0,221,168,128,215,189,
0,46,212
50010 DATA 128,118,141,0,52,165,0,19,64,53
50011 DATA 108,155,0,109,155,0,109,155,0,109,155,0,109,155,0,109,155,
0,105,27,0
50012 DATA 96,19,0,77,99,0,26,246,0,43,220,0,95,106,0,126,245,128,59,
239,0,110
50013 DATA 183,0,127,190,0,58,245,0,22,180
50014 DATA 61,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0
50015 DATA 0,0,0,254,0,0,254,0,0,254,0,0,254,0,0,12,0,0,12,0,0,12,0,
0,12,0,0,12
50016 DATA 57,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,6
50017 DATA 0,0,14,0,0,30,31,255,254,79,255,252,255,255,253,127,127,
142,0,63,192
50018 DATA 0,15,240
50019 DATA 28,30,0,0,127,0,0,255,128,0,255,128,0,201,128,0,73,0,0,42,
0,0,28,0,0
50020 DATA 20,0,0,28
50021 DATA 7,248,0,0,216,0,0,248
50022 DATA 0
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);:{left}"::ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}"::
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}"::
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296

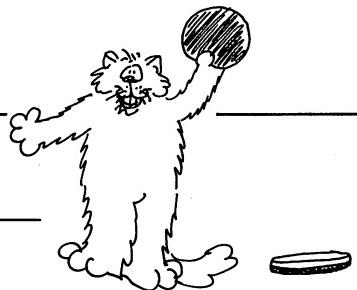
```

```
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
    TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$(" {21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$ :PRINT "{rvs-on}"
    TAB(TB)LEFT$(" {21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
    MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$ :PRINT "{7°down cyn 9°space}
(C)_1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$(" {pur cyn}",K+1)LEFT$(MS$,
    Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

REVERSI

(Uses joystick)

By Glen Fisher



Better known as "Othello," the game of REVERSI is an old game of strategy, invented by an Englishman in the 1800's. It is played on an 8×8 board with 64 pieces, with each piece white on one side and black on the other. The color on top shows who owns a piece at any given time. When a piece is captured, it is turned over to show the color of the person taking it.

When you start REVERSI, you tell the C-64 whether you're playing against it or against another person. Next you choose one of the two starting patterns for the first four pieces on the board. After the board is set up, each player takes turns placing a piece on the board. Opponent's pieces already on the board may be captured, causing them to be reversed. Once a piece is placed on the board it is never moved again, although it may be reversed any number of times. During the game, the C-64 displays the number of pieces owned by each player at the sides of the screen.

When it's your turn to play, a cursor the color of your pieces will appear on the board. Move it around using the joystick, and press the joystick button when you've selected your move. If the move is illegal, the cursor will reappear and you'll have to move elsewhere. If the move is legal, all captures will be made,

and your opponent will be given his or her turn. If you can't make any legal moves, press RETURN on the keyboard.

A legal move is any move that captures a piece, and any move that doesn't is illegal. During your move, you must capture at least one of your opponent's pieces. Captures are made by placing a piece in the same row, column, or diagonal as a piece you already own, with one or more of your opponent's pieces (but no empty squares) between them. The enemy pieces between your old piece and your newly placed piece (along rows, columns, or diagonals) are captured, and all are flipped to your color. They will now count as your pieces until your opponent takes them back. Sometimes, one move will capture pieces in more than one direction. Theoretically, you could capture pieces in eight directions at once, but it's not very common.

Note that captures only occur with a newly placed piece. So, if your opponent places a piece between two of yours, then it will stay his or hers, even though you have pieces on both sides of it. Also, a captured piece cannot itself be used to make more captures in the same turn. The game is over when no more pieces can be placed on the board. The player who has the most pieces on the board is the winner.

```
0 PG$="^R^E^V^E^R^S^I^":AU$="BY^GLEN^FISHER":JF=1
1 REM REVERSI -- BY GLEN FISHER
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/11/83 GF
90 GOTO 62000
100 DIM BD(9,9),JD(15):WH=1:BL=2:MT=0:POKE VIC+32,0:POKE VIC+33,6
101 FOR I=0 TO 15:READ JD(I):NEXT:DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,
     8,5
102 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
     16)=0
103 FOR I=0 TO 28:POKE SID+I,0:NEXT:POKE SID+6,240:POKE SID+4,129
104 POKE SID+24,15
```

```

110 DIM RC(7),CC(7)
120 FOR I=0 TO 7:READ RC(I),CC(I):NEXT I
130 DATA -1,-1,-1,0,-1,1,0,1
132 DATA 1,1,1,0,1,-1,0,-1
140 DIM DC(9):FOR I=1 TO 9:READ DC(I):NEXT I
150 DATA 6,5,4,7,-1,3,0,1,2
250 GOSUB 4000
260 PC$(MT)="-":PC$(BL)="{blk Q}":PC$(WH)="{yel Q}":PC(MT)=6:
    PC(WH)=15:PC(BL)=0
500 DIM VL%(7,7),V(4)
510 V(1)=2:V(2)=3:V(3)=1:V(4)=4
520 FOR I=0 TO 7:V=V(ABS(3.5-I)+.5)/2
530 FOR J=0 TO 7:VL%(I,J)=VL%(I,J)+V:NEXT J
540 FOR J=0 TO 7:VL%(J,I)=VL%(J,I)+V:NEXT J
550 NEXT I
560 GOSUB 4600
600 GOSUB 4700
700 FOR I=1 TO 8:FOR J=1 TO 8:BD(I,J)=0:NEXT J,I
710 BD(4,4)=WH:BD(4,5)=BL
720 BD(5,4)=N:BD(5,5)=3-N
900 GOSUB 50000:POKE VIC+39,6
910 FOR R=4 TO 5:FOR C=4 TO 5:GOSUB 39900
920 PRINT PC$(BD(R,C)):NEXT C:NEXT R
950 LR=3:HR=6:LC=3:HC=6:R=3:C=3
1000 M=5
1010 PRINT "{home 8°down cyn}":PRINT "{down}"
1020 PRINT TAB(7);PC$(WH);TAB(32);PC$(BL)
1030 WC=2:BC=2
1040 PRINT:PRINT TAB(6);"{cyn}";WC;TAB(31);BC
2000 PRINT "{home 8°down cyn}":CH=0
2005 GET T$:IF T$="Q" THEN 5000
2010 T=(M AND 1):IF T THEN 2100
2020 PRINT TAB(30);"MOVE"
2025 PL=BL:PL$=PC$(PL)
2030 ON NP+1 GOSUB 3500,3500,3000
2035 IF CH>0 THEN BC=BC+1
2040 GOTO 2200
2100 PRINT TAB(5);"MOVE"
2105 PL=WH:PL$=PC$(PL)
2110 ON NP+1 GOSUB 3500,3000,3000
2120 IF CH>0 THEN WC=WC+1
2200 PRINT "{home 8°down}":PRINT TAB(5);:IF (M AND 1)=0 THEN
    PRINT TAB(30);:CH=-CH
2210 PRINT "{4°space 3°down}"
2220 WC=WC+CH:BC=BC-CH
2230 PRINT TAB(6);"{cyn}";WC;"{left}~":"TAB(31);BC;"{left}":"
2240 IF CH=0 THEN SK=SK+1
2250 IF CH<>0 THEN SK=0

```

```
2260 M=M+1
2300 IF WC+BC=64 THEN 2500
2310 IF SK=2 THEN 2500
2320 IF WC=0 OR BC=0 THEN 2500
2330 IF CH=0 THEN 2000
2400 IF R<=LR AND LR>1 THEN LR=R-1
2410 IF R>=HR AND HR<8 THEN HR=R+1
2430 IF C<=LC AND LC>1 THEN LC=C-1
2440 IF C>=HC AND HC<8 THEN HC=C+1
2450 GOTO 2000
2500 POKE VIC+39,6
2505 IF WC=BC THEN PRINT "{home}";TAB(10);"THE GAME WAS A TIE.":GOTO 2600
2510 PRINT "{home 9°down}";TAB(4);:PL$="WHITE"
2520 IF BC>WC THEN PRINT TAB(29);:PL$="BLACK"
2530 PRINT "WINNER":PRINT "{home down}";TAB(10);PL$;" WINS THE GAME."
2600 PRINT:GOSUB 4500:IF YN THEN PRINT "{clr}":GOTO 600
2620 GOTO 5000
3000 GOSUB 39800
3010 RX=6+PC(PL)
3100 T=ABS(BD(R,C))+1:PR$=PC$(PL)
3110 RV=6:TM=0
3120 IF TI>TM THEN RV=RX-RV:POKE VIC+39,RV:TM=TI+30
3130 GET T$:DR=FNJ(0):T=FNB(0):IF T$="" AND DR=5 AND T=0 THEN 3120
3140 IF T$="Q" THEN 5000
3150 IF DR<>5 THEN 3200
3160 IF T THEN 3400
3170 IF T$=CHR$(13) THEN CH=0:RETURN
3180 GOTO 3120
3200 DR=DC(DR):IF DR<0 THEN 3120
3220 R=R+RC(DR):C=C+CC(DR)
3230 IF R<1 THEN R=8
3240 IF R>8 THEN R=1
3250 IF C<1 THEN C=8
3260 IF C>8 THEN C=1
3270 GOSUB 39800:GOTO 3100
3400 IF BD(R,C)<>MT THEN 3120
3410 POKE VIC+39,PC(3-PL)
3420 GOSUB 38200:IF CH=0 THEN GOSUB 39900:BD(R,C)=MT:PRINT ".{left}":GOTO 3110
3430 RETURN
3500 CH=0:RM=R:CM=C:MC=-1E20:O=0:POKE VIC+39,PC(PL)
3510 FOR R=LR TO HR
3520 FOR C=LC TO HC
3525 GET Q$:IF Q$="Q" THEN 5000
3530 IF BD(R,C)<>MT THEN 3610
3540 GOSUB 39800
```

```

3550 GOSUB 38000:IF CH=0 THEN 3610
3560 CH=CH+VL%(R-1,C-1)
3570 IF CH=MC THEN O=O+1:IF RND(1)<1/0 THEN RM=R:CM=C
3590 IF CH>MC THEN RM=R:CM=C:MC=CH
3610 NEXT C
3620 NEXT R
3800 R=RM:C=CM:IF MC<=0 THEN CH=0:RETURN
3810 IF R>HR OR C>HC THEN POKE VIC+1,0
3820 POKE VIC+39,PC(3-PL):GOSUB 39800:GOSUB 38200
3900 RETURN
4000 SB=13*64:SP=CRT+1016
4010 FOR I=0 TO 13:A=SB+3*I
4020 POKE A,255:POKE A+1,252:POKE A+2,0
4030 NEXT I
4035 FOR I=A+3 TO SB+63:POKE I,0:NEXT
4040 POKE SP,SB/64:POKE VIC+1,0:POKE VIC+16,0:POKE VIC+21,1
4050 POKE VIC+23,0:POKE VIC+29,0:POKE VIC+27,1:POKE VIC+39,14
4060 RETURN
4500 NO=0
4510 TM=0:SW=1:PRINT TAB(10);"PLAY AGAIN?blu  
YES{up}"  

4520 IF TI<TM THEN 4550
4530 PRINT TAB(23+NO*4);MID$("{wht blu}",SW,1);  
MID$("YESNO{shift-space}",3*NO+1,3);"up"  

4540 SW=3-SW:TM=TI+15
4550 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
4560 IF JB AND JD=15 THEN YN=(NO=0):GOSUB 4900:RETURN
4570 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 4510
4580 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 4510
4590 GOTO 4520
4600 NP=1:PRINT "{cyn}HOW MANY PEOPLE{6°left down 6°T up}  
ARE PLAYING?{down}"  

4610 TM=0:SW=1:PRINT "{blu 4°space}0{shift-space}.1..2{up}"  

4620 IF TI<TM THEN 4650
4630 PRINT TAB(3+NP*3);MID$("{wht blu}",SW,1);NP;"up"  

4640 SW=3-SW:TM=TI+15
4650 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
4660 IF JB AND JD=15 THEN PRINT TAB(3+NP*3);"";NP:GOSUB 4900:  
RETURN
4670 IF (JD AND 4)=0 AND NP>0 THEN NP=NP-1:GOTO 4610
4680 IF (JD AND 8)=0 AND NP<2 THEN NP=NP+1:GOTO 4610
4690 GOTO 4620
4700 W$=PC$(WH)+"{right}":B$=PC$(BL)+"{right}"
4705 G$="A*R*S down 5°left - right - right - down 5°left Q*+*W  
down 5°left - right - right - down 5°left Z*E*X 4°up 5°left}"  

4710 PRINT "{4°down cyn 4°space}START{down}"  

4715 PRINT "...{blu}";G$;"{down right}";W$;B$;"{2°down 4°left}";W$;  
B$;"{up cyn}{OR}"  


```

```
4720 PRINT "{2°up blu}";G$;"{down right}";W$;B$;"{2°down 4°left}";
      B$;W$;"{4°up}":PRINT TAB(2);
4800 N=INT(RND(1)*2)+1
4810 TM=0:SW=1:PRINT "{blu}";G$;"{up}":PRINT TAB(N*9-7);
4820 IF TI<TM THEN 4850
4830 PRINT MID$("wht blu",SW,1);G$;
4840 SW=3-SW:TM=TI+15
4850 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
4860 IF JB AND JD=15 THEN PRINT "{wht}";G$:GOSUB 4900:RETURN
4870 IF (JD AND 8)=0 AND N=1 THEN N=2:GOTO 4810
4880 IF (JD AND 4)=0 AND N=2 THEN N=1:GOTO 4810
4890 GOTO 4820
4900 IF (PEEK(JS) AND 16)=0 THEN 4900
4910 RETURN
4999 END
5000 POKE VIC+21,0:PRINT "{clr}":END
38000 CH=0:FOR D=0 TO 7
38010 RC=RC(D):CC=CC(D)
38020 GOSUB 39000:IF N>0 THEN CH=CH+N
38030 NEXT D
38040 RETURN
38200 GOSUB 39900:BD(R,C)=PL:PRINT PL$;"{left}";
38210 CH=0:FOR D=0 TO 7
38220 RC=RC(D):CC=CC(D)
38230 GOSUB 39000:GOSUB 39100:IF N>0 THEN CH=CH+N
38240 NEXT D
38250 RETURN
38300 GOSUB 39900:BD(R,C)=PL:PRINT PL$;"{left}";
38310 FOR D=0 TO 7
38320 RC=RC(D):CC=CC(D)
38330 N=OE(D):GOSUB 39400
38340 NEXT D
38350 RETURN
39000 R0=R:C0=C:REM COUNT
39010 FOR I=1 TO 8
39020 R0=R0+RC:C0=C0+CC:BD=BD(R0,C0)
39030 IF BD=PL THEN N=I-1:RETURN
39040 IF BD=MT THEN 39070
39050 NEXT I
39070 N=-1:RETURN
39090 :
39100 R0=R:C0=C:IF N<1 THEN RETURN
39110 FOR I=1 TO N:REM FLIP
39120 R=R+RC:C=C+CC
39130 BD(R,C)=3-BD(R,C)
39132 POKE SID+1,32
39135 GOSUB 39900
39136 PRINT PL$
```

```

39137 POKE SID+1,0
39140 NEXT I:R=R0:C=C0:RETURN
39190 :
39800 Z=RZ+16*R:IF PEEK(VIC+1)<>Z THEN POKE VIC+1,0
39810 POKE VIC,CZ+16*C:POKE VIC+1,Z:RETURN
39900 PRINT "{home}"
39910 PRINT LEFT$("{25°down}",2+R+R);
39920 PRINT LEFT$("{40°right}",10+C+C);
39930 RETURN
50000 PRINT "{clr 3°down blu}":TB=11:RZ=23+3*16:CZ=13+TB*8
50010 PRINT TAB(TB);"A*R*R*R*R*R*S""
50020 T$="- shift-space - shift-space - shift-space - shift-space -
shift-space - shift-space - shift-space - shift-space - shift-space -":
PRINT TAB(TB);T$
50030 FOR I=1 TO 7:PRINT TAB(TB);"Q*+*+*+*+*+*W":
PRINT TAB(TB);T$:NEXT I
50040 PRINT TAB(TB);"Z*E*E*E*E*E*E*X""
50050 RETURN
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{_}",ZC,1);"{left}";:ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$::PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$::PRINT "{7°down cyn 9°space}
(C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120

```

```
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

BOP (Uses joystick)

By Glen Fisher



This game is not fast moving or competitive, but it does give an interesting look at a different way of representing numbers. Before calculators and computers were invented, the Koreans invented "Chisanbop," a method of counting on your fingers. The fingers on the right hand are each worth one, and the thumb is worth five. The fingers on the left hand are worth ten each and the thumb 50. To show a number with your fingers, extend the fingers that add up to the number you want.

In our version, we've trapped someone inside the C-64, forcing him to press his hands against the glass. He is very obedient and will extend or fold down whichever fingers either you or the C-64 tell him to.

When you start, you'll be presented with a menu with five choices. The first four choices will be explained below; the fifth choice allows you to leave the program (and lets the little man rest). Once you've made a choice, you can get back to the menu by pressing RETURN.

The first choice, "SET UP FINGERS," instructs you to set the man's fingers to the number chosen by the C-64. If you get it wrong, BOP will tell you the

number that you've actually set. If you get it right, you'll be congratulated.

To set the fingers, use the small pointer on the screen which points to one finger. Move the pointer left or right by pushing the joystick in the direction you want to move. Extend a finger by pushing the joystick up, and fold it over by pushing the joystick down. When the fingers are set the way you think they should be, press the joystick button, and BOP will tell you how well you did.

The second choice, "READ THEM," arranges the fingers, and then you guess what number is shown. If you misread the fingers, the C-64 will tell you what the correct number was.

The third choice, "WATCH THEM," lets you pick a number, which the man will show on his fingers. Obviously, this is an excellent way to learn the Chisanbop system.

The fourth choice, "COUNT TO 99," puts the man through his paces as he counts from 0 to 99 with his fingers. Pressing RETURN will stop him and return you to the menu.

```
0 PG$="^B^O^P^":AU$="BY^GLEN^FISHER":JF=1
1 REM BOP -- BY GLEN FISHER
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/16/83 GF
90 GOTO 62000
100 DIM RC(9,1),UP(9),JD(15)
101 FOR I=0 TO 15:READ JD(I):NEXT:DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,
     8,5
102 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
     16)=0
110 FOR I=0 TO 9:READ RC(I,0):NEXT I
115 FOR I=0 TO 9:READ RC(I,1):NEXT I
120 DATA 7, 5, 5, 7, 9, 9, 7, 5, 5, 7
130 DATA 9,12,13,16,17,21,22,25,26,29
140 PT$="{MG- 2°N 2°M BHN}"
```

```

160 DN$="left 24°down"
190 ZER=0:RN=(RC(4,0)+RC(5,0))/2+1:CN=(RC(4,1)+RC(5,1))/2
210 CR$=CHR$(13)
500 PRINT "{clr}";
505 PRINT "{down}DO YOU WANT TO:"
510 PRINT ".1) SET UP FINGERS"
515 PRINT ".2) READ THEM"
520 PRINT ".3) PICK A NUMBER"
525 PRINT ".4) COUNT TO 99"
526 PRINT ".5) QUIT"
530 PRINT "ENTER YOUR CHOICE:":GOSUB 60000
540 IF IN$="1" THEN GOSUB 9000:GOTO 1000
545 IF IN$="2" THEN GOSUB 9000:GOTO 3500
550 IF IN$="3" THEN GOSUB 9000:GOTO 3000
555 IF IN$="4" THEN GOSUB 9000:GOTO 3700
556 IF IN$<>"5" THEN GOTO 560
557 PRINT "{clr down}":END
560 PRINT "{clr}PLEASE TYPE '1', '2', '3', '4', OR '5'.{down}"
570 GOTO 510
1000 NUM=INT(RND(1)*99)+1:RJ=0
1010 PRINT "{home}SET UP THE NUMBER";NUM;"{left}."
1030 NF=0
1050 PRINT "{home}";SPC(1+RC(NF,1));LEFT$(DN$,1+RC(NF,0));
1060 PRINT MID$(PT$,1+NF,1);"{left}";
1100 GET T$:IF T$<>"" THEN 500
1105 IF FNB(0) THEN PRINT "{shift-space}":GOTO 1200
1110 J=FNJ(0):IF J=5 THEN 1100
1115 PRINT ".{left}";
1120 IF J=4 THEN NF=NF-1:IF NF<0 THEN NF=9
1125 IF J=6 THEN NF=NF+1:IF NF>9 THEN NF=0
1130 IF J=8 AND UP(NF)=0 THEN GOSUB 4000
1135 IF J=2 AND UP(NF) THEN GOSUB 4000
1160 GOTO 1050
1200 N=0:FOR I=3 TO 0 STEP -1
1210 IF UP(I)=ZER THEN 1230
1220 N=N+10:NEXT I
1230 IF I<=0 THEN 1270
1240 FOR I=I TO 0 STEP -1
1250 IF UP(I)<>ZER THEN 1900
1260 NEXT I
1270 N=N+50*ABS(UP(4)<>ZER)
1280 FOR I=6 TO 9
1290 IF UP(I)=ZER THEN 1310
1300 N=N+1:NEXT I
1310 IF I>=9 THEN 1350
1320 FOR I=I TO 9
1330 IF UP(I)<>ZER THEN 1900
1340 NEXT I

```

```

1350 N=N+5*ABS(UP(5)<>ZER)
1360 IF N=NUM THEN PRINT "{home down}RIGHT!";:GOTO 1500
1370 PRINT "{home down}NO.";:GOTO 1500
1500 PRINT ".THE NUMBER ON MY FINGERS IS";N;"{left}."
1510 FOR I=1 TO 2000:NEXT I
1520 GOSUB 5500
1530 GOTO 1000
1900 PRINT "{home down}THAT ISN'T EVEN A NUMBER!":GOTO 1510
2000 T=INT(RND(1)*1024)
2010 FOR NF=0 TO 9
2030 UP=1 AND T:T=INT(T/2)
2040 IF UP(NF)<>UP THEN GOSUB 4000
2050 NEXT NF
2060 GOTO 2000
2990 FOR I=1 TO 1000:NEXT I
3000 GOSUB 7000
3010 NUM=NM:IF NM<0 THEN 500
3020 GOSUB 6000
3030 FOR NF=0 TO 9
3050 IF UP(NF)<>F(NF) THEN GOSUB 4000
3060 NEXT NF
3070 PRINT "{home}";SPC(CN-(NM>9));LEFT$(DN$,RN);NUM;"{left}."
3080 GOTO 3000
3500 NUM=INT(RND(1)*100):NM=NUM:GOSUB 6000
3510 FOR NF=0 TO 9
3530 IF UP(NF)<>F(NF) THEN GOSUB 4000
3540 NEXT NF
3550 GOSUB 7000:IF NM<0 THEN 500
3560 IF NM=NUM THEN PRINT "{home down}RIGHT!";:GOTO 3580
3570 PRINT "{home down}NO.";
3580 PRINT ".THE NUMBER IS";NUM;"{left}."
3590 FOR I=1 TO 2000:NEXT I
3600 GOTO 3500
3700 FOR NUM=0 TO 99:NM=NUM:GOSUB 6000
3701 GET T$:IF T$<>"" THEN 500
3705 PRINT "{home}";SPC(CN-(NM>9));LEFT$(DN$,RN);NUM;"{left}."
3710 FOR NF=0 TO 9
3720 PRINT "{home}";SPC(1+RC(NF,1));LEFT$(DN$,1+RC(NF,0));
3730 IF UP(NF)<>F(NF) THEN GOSUB 4000
3740 NEXT NF
3750 FOR I=1 TO 500:NEXT I
3760 NEXT NUM
3770 GOTO 500
4000 PRINT "{home red}";SPC(1+RC(NF,1));LEFT$(DN$,1+RC(NF,0));
4010 ON NF+1 GOSUB 5000,5010,5020,5030,5040,5050,5060,5070,5080,5090
4020 PRINT "{rvs-off wht}";:RETURN
5000 PRINT "{2°right down}";:IF UP(0) THEN PRINT ".{left 2°down
rvs-on @}";:GOTO 5003

```

```

5002 PRINT "{rvs-on M left 2°down}";MID$("{M}^",1+UP(1),1);
5003 PRINT "{left 3°up 2°left}";:UP(0)=1-UP(0)
5005 RETURN
5010 PRINT "2°down";
5011 IF UP(1) THEN PRINT ".{left down}.{left 2°down left rvs-on}";
MID$("{@M}",1+UP(0),1);>{@};:GOTO 5013
5012 PRINT "{rvs-on M left down M left 2°down left}";MID$("{@}^",
1+UP(0),1);MID$("{M}^",1+UP(2),1);
5013 PRINT "{left 5°up}";:UP(1)=1-UP(1)
5015 RETURN
5020 PRINT "2°down";
5021 IF UP(2) THEN PRINT ".{left down}.{left 2°down left rvs-on}";
MID$("{@M}",1+UP(1),1);{@};:GOTO 5023
5022 PRINT "{rvs-on M left down M left 2°down left}";MID$("{@}^",
1+UP(1),1);MID$("{@PM}^",1+UP(3)+2*UP(4),1);
5023 PRINT "{left 5°up}";:UP(2)=1-UP(2)
5025 RETURN
5030 PRINT "2°left down";:IF UP(3)=0 THEN 5032
5031 PRINT ".{left 2°down left rvs-on}";MID$("{@M}",1+(UP(2) AND
UP(4)),1);{@};:GOTO 5033
5032 PRINT "{rvs-on M left 2°down left}";MID$("{@P@}^",
1+UP(2)+2*UP(4),1);{@};
5033 PRINT "{right 3°up}";:UP(3)=1-UP(3)
5035 RETURN
5040 PRINT "4°left down rvs-on";:IF UP(4)=0 THEN 5042
5041 PRINT MID$("{@P}",1+(UP(2) AND UP(3)),1);{@ left down 2°M left
down M rvs-off}.{left up}.;:GOTO 5043
5042 PRINT MID$("{@M@}^",1+UP(2)+2*UP(3),1);{@ left down}.{M left
down}.{rvs-off £ left up rvs-on £ rvs-off £};
5043 PRINT "{down}";:UP(4)=1-UP(4)
5045 RETURN
5050 PRINT "2°down right";:IF UP(5)=0 THEN 5052
5051 PRINT ".{left down}.{rvs-on N left up 2°N left up}";
MID$("{LP}",1+(UP(6) AND UP(7)),1);:GOTO 5053
5052 PRINT "{* rvs-on * rvs-off left down * rvs-on}.{left up G}
.{left up}";MID$("{2°L G}^",1+UP(6)+2*UP(7),1);
5053 PRINT "2°down 5°left";:UP(5)=1-UP(5)
5055 RETURN
5060 PRINT "2°right down";
5061 IF UP(6) THEN PRINT ".{left 2°down rvs-on L}";MID$("{LG}",
1+(UP(5) AND UP(7)),1);:GOTO 5063
5062 PRINT "{rvs-on G left 2°down G}";MID$("{2°L P}^",
1+UP(5)+2*UP(7),1);
5063 PRINT "2°left 3°up 2°left";:UP(6)=1-UP(6)
5065 RETURN
5070 PRINT "2°down";
5071 IF UP(7) THEN PRINT ".{left down}.{left 2°down rvs-on L}";
MID$("{LG}",1+UP(8),1);:GOTO 5073

```

```

5072 PRINT "{rvs-on G left down G left 2°down}";MID$("LG",1+UP(5)+2*UP(6),1);MID$("L",1+UP(8),1);
5073 PRINT "2°left 5°up";:UP(7)=1-UP(7)
5075 RETURN
5080 PRINT "2°down";
5081 IF UP(8) THEN PRINT ".{left down}.{left 2°down rvs-on L}";
MID$("LG",1+UP(9),1);:GOTO 5083
5082 PRINT "{rvs-on G left down G left 2°down}";MID$("G",1+UP(7),
1);MID$("L",1+UP(9),1);
5083 PRINT "2°left 5°up";:UP(8)=1-UP(8)
5085 RETURN
5090 PRINT "2°left down";
5091 IF UP(9) THEN PRINT ".{left 2°down rvs-on L}";:GOTO 5093
5092 PRINT "{rvs-on G left 2°down}";MID$("G",1+UP(8),1);
5093 PRINT "right 3°up";:UP(9)=1-UP(9)
5095 RETURN
5500 PRINT "home rvs-off";
5510 FOR II=1 TO 3
5520 PRINT "39°space"
5530 NEXT II
5540 PRINT "home";:RETURN
6000 REM
6010 FOR I=0 TO 9:F(I)=ZER:NEXT I
6020 T=INT(NM/5):N=NM-5*T:NM=T
6030 IF N>0 THEN FOR I=6 TO 5+N:F(I)=1-F(I):NEXT I
6040 T=INT(NM/2):N=NM-2*T:NM=T
6050 IF N THEN F(5)=1-F(5)
6060 T=INT(NM/5):N=NM-5*T:NM=T
6070 IF N>0 THEN FOR I=3 TO 4-N STEP -1:F(I)=1-F(I):NEXT I
6080 T=INT(NM/2):N=NM-2*T:NM=T
6090 IF N THEN F(4)=1-F(4)
6100 RETURN
7000 GOSUB 5500
7100 PRINT "home NUMBER?";:GOSUB 60000
7105 IF IN$="" THEN NM=-1:RETURN
7110 IF IN$="" THEN 7300
7120 Z=0:FOR I=1 TO LEN(IN$)
7130 :T$=MID$(IN$,I,1):IF T$="-" THEN 7160
7135 :IF T$>="0" AND T$<="9" THEN Z=Z+1:GOTO 7160
7140 IF T$"." THEN PRINT "home down NO FRACTIONS, PLEASE.":GOTO 7310
7150 GOTO 7300
7160 NEXT I:IF Z=0 THEN 7300
7170 NM=VAL(IN$)
7180 IF NM<0 THEN PRINT "home down TOO SMALL.":GOTO 7310
7190 IF NM>99 THEN PRINT "home down TOO BIG.":GOTO 7310
7200 RETURN
7300 PRINT "home down THAT'S NOT A NUMBER."

```

```

7310 FOR I=1 TO 1000:NEXT I
7320 GOTO 7000
9000 PRINT "{clr red 12°right 7°down}";
9010 PRINT "{rvs-on 2°M 3°left down 4°M 4°left down 4°M}";
9020 PRINT "{4°left down}^{M 4°left down}^{M£ rvs-off £}";
9030 PRINT "{6°left down rvs-on 4°space rvs-off £ 5°left down}";
9040 PRINT "{_* rvs-on}^{rvs-off £ 3°left down rvs-on}^{2°left
down}^{";
9050 PRINT "{8°up 11°right}";
9060 PRINT "{rvs-on 2°G 3°left down 4°G 4°left down 4°G}";
9070 PRINT "{4°left down G}^{6°left down rvs-off * rvs-on *G}^{";
9080 PRINT "{5°left down rvs-off * rvs-on 4°space 4°left down}";
9090 PRINT "{rvs-off * rvs-on}^{rvs-off £ 3°left down rvs-on}
^{2°left
down}^{rvs-off wht}";
9100 FOR II=0 TO 9:UP(II)=1:NEXT II
9110 RETURN
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);"{left}";:ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
(C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
Q)

```

```
62100 Q=Q+1:IF Q>LEN(M$) THEN Q=0:K=1-K  
62110 GOTO 62070  
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

SPOT

By Art Carpet



SPOT is a simple two-player game similar to tic-tac-toe but a bit more difficult. You must get four squares in a line to win; the line may be horizontal, vertical, or diagonal. However, you can't just put your squares anywhere. After selecting the column where you want your square to go, the square will drop down to the bottom of that column. Therefore, when you want to put a square several rows up, you will have to fill up the column below it

first. You will find that building up a column is tricky business, since the squares you stack up may help your opponent to win.

One player's squares are shown in green and the other's in yellow. The C-64 will tell you when it is your turn, and you choose a column by pressing the corresponding number. Your square will drop into that column. When you win, by getting four squares in a row, the C-64 will flash the winning squares.

```
0 PG$="{shift-space}S~P~O~T~":"AU$="BY~ART~CARPET"
1 REM SPOT -- BY ART CARPET
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
8 :
10 REM AS OF 8/11/83 GF
90 GOTO 62000
100 BL$="{39° space}":GOSUB 1340
190 V=0:PRINT "{clr 2°down cyn 12°space}1~~~2~~~3~~~4~~~5~~~6~~~7"
200 A$="{10°space A 3°* R 3°* R 3°* R 3°* R 3°* R 3°* S}"
210 B$="{10°space - 3°shift-space - 3°shift-space - 3°shift-space
      - 3°shift-space - 3°shift-space - 3°shift-space -
      3°shift-space - }"
220 C$="{10°space Q 3°* + 3°* + 3°* + 3°* + 3°* + 3°* W}"
230 D$="{10°space Z 3°* E 3°* E 3°* E 3°* E 3°* E 3°* X}"
240 PRINT A$:FOR T=1 TO 5:PRINT B$:PRINT C$:NEXT
250 PRINT B$:PRINT D$:IF GA<>0 THEN 290
260 DIM A(7,6),B(42)
270 A$(1)="{rvs-on yel}~~~{down 3°left}~~~{wht rvs-off}":
   A$(2)="{grn rvs-on}~~~{down 3°left}~~~{rvs-off wht}":
   A$(3)="~~~{down 3°left}~~~"
280 FOR T=1 TO 42:READ Z:B(T)=Z+32768:NEXT
290 GA=1:FOR T=1 TO 6:FOR Y=1 TO 7:V=V+1:A(Y,T)=B(V):NEXT:NEXT:V=0
300 X=INT(RND(TI)*2+1):AT=0
310 J=0:POKE QL,24:PRINT "{up}";BL$
320 GOSUB 2000
330 POKE QL,20:PRINT "{up 3°right}";A$(X);"{2°down}":
   PRINT N$(X);"'S~MOVE";:
340 GOSUB 2100:IF R$="Q" THEN PRINT "{clr}":END
```

```

360 IF VAL(R$)<1 OR VAL(R$)>7 THEN 340
370 PRINT:POKE QL,24:PRINT "{up}";BL$:POKE QL,20:PRINT "{up}";A$(3)
380 M=VAL(R$)
390 FOR T=1 TO 6
400 IF A(M,T)<9 THEN J=J+1
410 IF J=6 THEN 310
420 NEXT:J=0
430 AT=AT+1
440 FOR T=6 TO 1 STEP -1
450 IF A(M,T)>8 THEN 470
460 NEXT
470 V=M*4+7:H=T*3+2
480 FOR K=1 TO 2:TW=TI
490 POKE QL,H:PRINT "{up}" TAB(V)A$(X)
500 IF TI<TW+10 THEN 500
510 POKE QL,H:PRINT "{up}" TAB(V)A$(3)
520 IF TI<TW+20 THEN 520
530 NEXT:POKE QL,H:PRINT "{up}" TAB(V)A$(X)
540 A(M,T)=X↑3
550 REM CHECK FOR HORIZ WIN
560 FOR C=1 TO 4
570 IF A(C,T)+A(C+1,T)+A(C+2,T)+A(C+3,T)=4*(X↑3) THEN 590
580 NEXT:GOTO 640
590 M(1)=C:T(1)=T
600 M(2)=C+1:T(2)=T
610 M(3)=C+2:T(3)=T
620 M(4)=C+3:T(4)=T
630 GOTO 950
640 REM VERT WIN?
650 FOR C=1 TO 3
660 IF A(M,C)+A(M,C+1)+A(M,C+2)+A(M,C+3)=4*(X↑3) THEN 680
670 NEXT:GOTO 730
680 M(1)=M:T(1)=C
690 M(2)=M:T(2)=C+1
700 M(3)=M:T(3)=C+2
710 M(4)=M:T(4)=C+3
720 GOTO 950
730 REM L TO R DIAG?
740 FOR K=1 TO 4
750 FOR C=6 TO 4 STEP -1
760 IF A(K,C)+A(K+1,C-1)+A(K+2,C-2)+A(K+3,C-3)=4*(X↑3) THEN 780
770 NEXT:NEXT:GOTO 830
780 M(1)=K:T(1)=C
790 M(2)=K+1:T(2)=C-1
800 M(3)=K+2:T(3)=C-2
810 M(4)=K+3:T(4)=C-3
820 GOTO 950
830 REM CHECK FOR R TO L DIAG

```

```

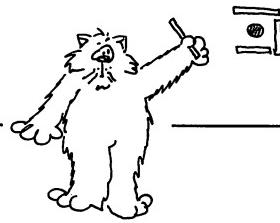
840 FOR K=7 TO 4 STEP -1
850 FOR C=6 TO 4 STEP -1
860 IF A(K,C)+A(K-1,C-1)+A(K-2,C-2)+A(K-3,C-3)=4*(X↑3) THEN 880
870 NEXT:NEXT:GOTO 930
880 M(1)=K:T(1)=C
890 M(2)=K-1:T(2)=C-1
900 M(3)=K-2:T(3)=C-2
910 M(4)=K-3:T(4)=C-3
920 GOTO 950
930 IF AT=42 THEN 1036
940 X=3-X:GOTO 310
950 POKE QL,23:PRINT "{up}";TAB(10)N$(X) "IS THE WINNER!"
960 FOR I=1 TO 5
970 FOR Z=1 TO 4
980 M=M(Z):T=T(Z)
990 V=M*4+7:H=T*3+2
1000 POKE QL,H:PRINT "{up}" TAB(V)A$(3)
1010 FOR P=1 TO 100:NEXT
1020 POKE QL,H:PRINT "{up}" TAB(V)A$(X)
1030 NEXT:NEXT:GOTO 1040
1036 POKE QL,23:PRINT "{up}STALEMATE!~"
1040 POKE QL,24:PRINT "{clr 3°down}WANT TO PLAY AGAIN?~";:V=0:
GOSUB 60000
1050 IF LEFT$(IN$,1)<>"N" THEN GA=1:PRINT "{clr}":GOTO 190
1060 PRINT "{clr 2°down}":END
1080 DATA 171,175,179,183,187,191,195,291,295,299,303,307,311,315,
411,415
1090 DATA 419,423,427,431,435,531,535,539,543,547,551,555,651,655,
659,663
1100 DATA 667,671,675,771,775,779,783,787,791,795
1340 PRINT "{clr 3°down}WHO'S PLAYER ONE?~";:GOSUB 60000:N$(1)=IN$
1345 IF IN$="" THEN N$(1)="PLAYER 1":PRINT TAB(21);"{up}";N$(1)
1350 PRINT "{2°down}WHO'S PLAYER TWO?~";:GOSUB 60000:N$(2)=IN$
1355 IF IN$<>"" THEN RETURN
1360 N$(2)="PLAYER 2":PRINT TAB(21);"{up}";N$(2):FOR I=1 TO 500:
NEXT:RETURN
2000 GET Z$:IF Z$<>"" THEN 2000
2010 RETURN
2100 ZT=TI:ZC=2
2110 GET R$:IF R$<>"" THEN 2140
2120 IF ZT<=TI THEN PRINT MID$("~?",ZC,1);"{left}";:ZC=3-ZC:ZT=TI+15
2130 GOTO 2110
2140 RETURN
60000 IN$=~":ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);"{left}";:ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010

```

```

60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT ".{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$ :PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$ :PRINT "{7°down cyn 9°space}
(C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```



DOTS (Uses joystick)

By Glen Fisher

This is a computerized form of a game that just about everyone has played at one time or another. A 6×6 grid of dots is drawn, each dot at the corner of a square. (DOTS actually shows parts of the lines connecting the dots as well.) You and the computer take turns connecting the dots with lines. Lines can go only along the edge of a square and can't go through or past a dot. When a square is boxed in by the lines, the player who drew the last line gets the square and marks it. Your boxes are marked with an "O" and the computer's with an "X". The game is over when no more lines can be drawn, and the winner is the player with the most squares.

In this version, you draw a line by using the joy-

stick to move a line-shaped cursor around the board. When you've placed the line where you want it, press the joystick button. To change the line from horizontal to vertical (or vice versa), you move the joystick diagonally. When it's not your turn, the C-64 will display a gently undulating line so you'll know you're not forgotten, and you'll accept your losses more calmly.

You can change the size of the board by changing the variables RS and CS at line 130. RS is the number of rows of squares. There can't be more than 11 rows or 19 columns of squares. However, the bigger the board, the longer the computer takes to make its move.

```
0 PG$="^D^O^T^S^":AU$="BY^GLEN^FISHER":JF=1
1 REM DOTS -- BY GLEN FISHER
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
8 :
10 REM AS OF 8/11/83 GF
90 GOTO 62000
100 POKE VIC+32,6:POKE VIC+33,6:DIM JD(15)
101 FOR I=0 TO 15:READ JD(I):NEXT:DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,
8,5
102 JS=56320:DEF FNJ(X)=JD(PEEK(JS+X) AND 15):
DEF FNB(X)=(PEEK(JS+X) AND 16)=0
110 T=RND(-TI)
120 TBA=0:HZ=1:VT=2:HU=1:CO=2
130 RS=6:CS=6:SQ=RS*CS
140 RF=11-RS:CF=19-CS
200 R=RS+2:C=CS+2:DIM LC(R,C),VH(R,C),RT(4,4)
210 T=RND(-TI)
220 FOR I=0 TO 3:FOR J=0 TO 3:READ RT(I,J):NEXT J:NEXT I
230 DATA 3,4,2,6,4,5,1,7,2,1,0,8,6,7,8,9
240 DIM SV(1):SV(0)=0:SV(1)=1
250 Q$="^3^Y ED*FR ^3^P RF*DE":Q$=Q$+Q$:Q$=Q$+Q$
260 FOR I=1 TO 9:READ RM(I),CM(I):NEXT I
270 DATA 1,-1,2,0,1,1,0,-2,0,0,2,-1,-2,0,-1,1
```

```

280 GOSUL 2000
900 GOSUB 7000
910 PL=HU:SV(0)=0:SV(1)=1
990 IF PL=HU THEN GOSUB 1500:GOTO 1010
1000 GOSUB 5000
1010 GOSUB 5500:GOSUB 5600
1020 GOSUB 5800:IF N=0 THEN PL=3-PL
1030 IF SQ>0 THEN 990
1040 IF HQ>CQ THEN 1070
1050 IF HQ=CQ THEN 1080
1060 PRINT "{home}I WON, ^WITH ";CQ;"SQUARES ^TO ^YOUR ";HQ;:GOTO 1100
1070 PRINT "{home}YOU WON, ^WITH ";HQ;"SQUARES ^TO ^MY ";CQ;:GOTO 1100
1080 PRINT "{home}WE ^TIED, ^WITH ";HQ;"SQUARES ^EACH ^";
1100 PRINT "{left}."
1105 PRINT "{38° space up}"
1110 PRINT "WANT ^TO ^PLAY ^AGAIN? ^";:GOSUB 60000:T$=IN$
1120 IF LEFT$(T$,1)="Y" THEN 1200
1130 IF LEFT$(T$,1)<>"N" THEN PRINT "{up}";:GOTO 1105
1140 END
1200 FOR I=0 TO RS+2:FOR J=0 TO CS+2:LC(I,J)=0:VH(I,J)=0:NEXT J:
NEXT I
1210 SQ=RS*CS:CQ=0:HQ=0
1290 GOTO 900
1500 R=SV(0):C=SV(1):TM=0:SW=1:DL=0
1510 GOSUB 9000:GOSUB 9100
1525 T=FNJ(0):IF T=5 THEN DL=0:GOTO 1535
1530 IF TI>DL THEN 1600
1535 IF FNB(0) THEN POKE VIC+39,6:GOTO 1700
1540 GET T$:IF T$="Q" THEN 1800
1545 IF TI<TM THEN 1525
1550 POKE VIC+39,SW
1560 TM=TI+15:SW=7-SW:GOTO 1525
1600 DL=TI+15:SW=1:TM=0
1610 R=R+RM(T):C=C+CM(T)
1620 IF R<0 OR R>2*RS OR C<0 OR C>2*CS THEN R=R-RM(T):C=C-CM(T)
1630 IF FNB(0)=0 THEN 1510
1700 REM
1710 ROW=INT(R/2)+1:COL=INT(C/2)+1
1720 V=(R AND 1):H=1-V:DIR=HZ:IF V THEN DIR=VT
1730 IF VH(ROW,COL) AND (VT*V+HZ*H) THEN 1530
1740 SV(0)=R:SV(1)=C:POKE VIC+1,0
1750 RETURN
1800 POKE VIC+21,0
1810 PRINT "{home 22°down}";
1820 END
2000 SP=CRT+1016:R0=49+8*RF:C0=23+8*CF
2010 SB=13*64
2020 FOR I=0 TO 127:POKE SB+I,0:NEXT

```

```

2030 FOR I=9 TO 20:READ T:POKE SB+I,T:NEXT
2040 FOR I=0 TO 27 STEP 3:READ T:POKE SB+64+I,T:NEXT
2050 DATA 255,192,0,0,0,0,0,0,255,192,0
2060 DATA 18,18,18,18,18,18,18,18,18,18
2070 POKE VIC,0:POKE VIC+39,1
2080 POKE VIC+21,1
2090 RETURN
5000 MX=-1:DIR=TBA:ROW=COL=-1
5005 R=2*RS+1:C=0
5010 FOR I=1 TO RS
5020 FOR J=1 TO CS+1
5025 GOSUB 5200
5030 IF VH(I,J) AND VT THEN 5070
5040 F=RT(LC(I,J-1),LC(I,J))
5050 IF F<MX OR (F=MX AND RND(1)>.5) THEN 5070
5060 DIR=VT:MX=F:ROW=I:COL=J
5070 NEXT J
5080 NEXT I
5110 FOR I=1 TO RS+1
5120 FOR J=1 TO CS
5125 GOSUB 5200
5130 IF VH(I,J) AND HZ THEN 5170
5140 F=RT(LC(I-1,J),LC(I,J))
5150 IF F<MX OR (F=MX AND RND(1)>.5) THEN 5170
5160 DIR=HZ:MX=F:ROW=I:COL=J
5170 NEXT J
5180 NEXT I
5185 GOSUB 9000:PRINT "{39°space}";
5190 RETURN
5200 GOSUB 9000:PRINT LEFT$(Q$,2*CS+1);
5210 Q$=MID$(Q$,2,255)+LEFT$(Q$,1)
5220 RETURN
5390 :
5400 TJ=TI+DL
5410 IF TI<TJ THEN 5410
5420 RETURN
5490 :
5500 LC(ROW,COL)=LC(ROW,COL)+1
5510 IF DIR<>HZ THEN 5540
5520 LC(ROW-1,COL)=LC(ROW-1,COL)+1
5530 VH(ROW,COL)=VH(ROW,COL) OR HZ:RETURN
5540 LC(ROW,COL-1)=LC(ROW,COL-1)+1
5550 VH(ROW,COL)=VH(ROW,COL) OR VT:RETURN
5590 :
5600 R=(ROW-1)*2:C=(COL-1)*2
5605 IF (R AND 1) OR (C AND 1) THEN STOP
5610 GOSUB 9000
5620 T$="{down -}":IF DIR=HZ THEN T$="{right *}"

```

```

5630 PRINT T$;:IF PL=HU THEN RETURN
5640 DL=5:FOR QQ=0 TO 6
5650 GOSUB 5400:PRINT "{left}~";
5660 GOSUB 5400:PRINT "{left}";RIGHT$(T$,1);
5670 NEXT QQ:RETURN
5690 :
5700 R=R*2-1:C=C*2-1:GOSUB 9000
5710 SQ=SQ-1
5720 IF PL=HU THEN PRINT "{wht W wht}";:HQ=HQ+1:RETURN
5730 PRINT "{cyn V wht}";:CQ=CQ+1:RETURN
5790 :
5800 N=0
5810 R=ROW:C=COL:GOSUB 5900
5820 R=ROW-1:C=COL:GOSUB 5900
5830 R=ROW:C=COL-1:GOSUB 5900
5840 RETURN
5890 :
5900 IF LC(R,C)<>4 OR VH(R,C) AND 12 THEN RETURN
5910 VH(R,C)=VH(R,C) OR 12*PL:N=N+1
5920 GOSUB 5700:RETURN
5990 :
6000 PRINT "{clr}";
6010 FOR I=1 TO CS:PRINT "{R}~";:NEXT I:PRINT "{S}"
6020 PRINT "{up A down}"
6030 FOR J=2 TO RS
6040 FOR I=1 TO CS:PRINT "{+}~";:NEXT I:PRINT "{W}"
6050 PRINT "{up Q down}"
6060 NEXT J
6070 FOR I=1 TO CS:PRINT "{E}~";:NEXT I:PRINT "{X}"
6080 PRINT "{up Z home}";
6090 RETURN
6900 END
7000 PRINT "{clr}":L$="{A}~":"R$="{S}~":"M$="{R}~"
7010 FOR I=2 TO RS:L$=L$+"{Q}~":"R$=R$+"{W}~":"M$=M$+"{+}~":"NEXT I
7020 L$=L$+"{Z}~":"M$=M$+"{E}~":"R$=R$+"{X}~"
7030 FOR R=0 TO 2*RS STEP 2
7040 C=0:GOSUB 9000:PRINT MID$(L$,R+1,2);
7050 FOR C=2 TO 2*CS-2 STEP 2:PRINT MID$(M$,R+1,2);:NEXT C
7060 PRINT MID$(R$,R+1,1)
7070 NEXT R
7080 RETURN
8990 :
9000 PRINT "{home}";:R=R+RF:C=C+CF
9010 IF R THEN PRINT LEFT$("24°down",R);
9020 IF C THEN PRINT LEFT$("39°right",C);
9030 R=R-RF:C=C-CF
9040 RETURN
9100 POKE VIC,C0+8*C:POKE VIC+1,R0+8*R

```

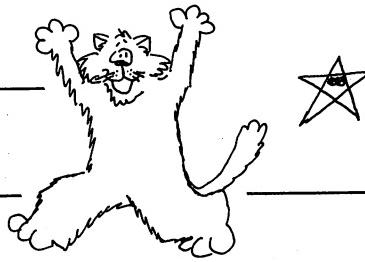
```

9120 POKE SP,14-(C AND 1)
9130 RETURN
50000 PRINT "{down}PRESS_{rvs-on}RETURN{rvs-off}_{TO}_CONTINUE:{"
50010 GET Z$:IF Z$="" THEN 50010
50020 RETURN
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);"{left}";:ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT_{RETURN}_{TO}_BEGIN":IF JF THEN
MS$="HIT_{JOYSTICK}_{BUTTON}_{TO}_BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
(C)_{1983}_{THE}_{CODE}_{WORKS}"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

CAPTURE (Uses joystick)

By Malcolm Michael



CAPTURE is deceptively simple. First, the screen is covered with numerous obstacles. Next, two beasts appear on the screen, depicted as purple stars. When they appear, you hear a brief noise, and they flash on and off a few times so you can see where they are located. Finally, you are placed on the screen in the form of a yellow ring.

The two beasts chase you, while you try to avoid them yet trap them at the same time. You are able to

move the obstacles around and form barriers that become cages where you trap the beasts. But it's not easy! They are smart little devils and can track your position remarkably well. If you are clever, you'll be able to build a cage leaving one opening and then lure them inside, while you sneak out and close the opening off. Should you be unfortunate and get too close to a beast, it will leap on you and—well, is your life insurance up-to-date?

```
0 PG$=" {shift-space}C^A^P^T^U^R^E^":AU$="BY^MALCOLM^MICHAEL":JF=1
1 REM CAPTURE -- BY MALCOLM MICHAEL
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 8/12/83 BS
90 GOTO 62000
100 Y=87:W=160:M=42:G=32:BR=214:DI=CM-CRT:DIM JD(15):Cl=8:C2=7:C3=4
101 FOR I=0 TO 15:READ JD(I):NEXT:DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,
     8,5
102 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
     16)=0
103 FOR I=0 TO 28:POKE SID+I,0:NEXT:POKE SID+6,240:POKE SID+4,17:
     VL=SID+24
110 UL=CRT+WD*2:S1=SID+1:S2=1E20:S3=1E20:REM S2=TONE,S3=CTRL
120 GOSUB 7000:DF=9-G:G=32
130 IF DF<0 OR DF>8 THEN 120
160 YS=0
170 BS=0
180 B=1
190 N=200
220 A=0:C=0:X=0
240 DIM BR(B),BC(B),BP(B),BM(B)
250 FOR A=1 TO 9:READ DR(A),DC(A):NEXT A
260 DATA 1,-1,1,0,1,1,0,-1,0,0,0,1,-1,-1,-1,0,-1,1
270 BL$=" {pur 38°space}"
290 PRINT "{clr}"
300 PRINT "{rvs-on}";BL$;"~{left inst}~"
310 FOR I=1 TO 21:PRINT "{rvs-on}~{rvs-off}";BL$;"{left rvs-on}
     ~{left inst}":NEXT I
```

```

320 PRINT "{rvs-on}";BL$;`{left inst}`{home}`
330 PRINT "{wht}`:FOR A=1 TO N
340 GOSUB 1130:POKE VP, BR
350 NEXT A
360 POKE S1, 50
370 FOR A=0 TO B:BM(A)=INT(RND(1)*DF)
375 GOSUB 1130:FOR I=-1 TO 1:FOR J=-1 TO 1:IF PEEK(VP+I*WD+J)=G
    THEN 390
380 NEXT J,I:GOTO 375
390 POKE VL,15:POKE S1,75+25*A:FOR C=1 TO 20:POKE VP,M+128
395 BP(A)=VP:BR(A)=VR:BC(A)=VC
400 POKE VP,M:POKE VP+DI,C1:NEXT C,A
410 GOSUB 1130:HP=VP:HR=VR:HC=VC
420 POKE VL,0
430 FOR A=0 TO B
440 IF ABS(HR-BR(A))<10 AND ABS(HC-BC(A))<20 THEN 410
450 NEXT A
460 POKE VL,15
470 POKE HP,Y:POKE HP+DI,C2
480 PRINT "{home 6°space}`MOVE`WHEN`READY...`"
490 POKE S1,255:POKE HP,(PEEK(HP)+128) AND 255:FOR A=1 TO 200:NEXT:
X=FNJ(0)
500 POKE S1,245:IF X=5 THEN 490
510 PRINT "{home}`;BL$
520 GOSUB 1110:TI$="000000":POKE HP,Y:POKE HP+DI,C2
530 I=0:POKE VL,0:GOTO 550
540 X=FNJ(0):BM=0:GET Q$:IF Q$="Q" THEN PRINT "{clr 2°down}`:END
550 PRINT "{home}`TIME:`;"MID$(TI$,3,2);":";RIGHT$(TI$,2)
560 IF A=5 THEN 590
570 OCP=CP:CR=DR(X):CC=DC(X):CP=CR*WD+CC
580 IF OCP<>CP THEN LL=0
590 SD=0
600 X=PEEK(HP+LL+CP):IF X=BR THEN LL=LL+CP:SD=16:GOTO 600
610 POKE VL,SD:POKE S1,128:IF CP=0 THEN 670
620 IF X=W OR (X=M AND LL<>0) THEN I=0:GOTO 670
640 HP=HP+CP:IF LL<>0 THEN POKE HP+LL, BR
650 POKE HP-CP,G:POKE HP-CP+DI,C3:POKE HP,Y:POKE HP+DI,C2:
    IF X=M THEN 820
660 HR=HR+CR:HC=HC+CC
670 K=0:FOR A=0 TO B:IF BP(A)>0 THEN 675
672 IF ABS(BR(A)-HR)>1 OR ABS(BC(A)-HC)>1 THEN 790
673 BP(A)=ABS(BP(A))
675 BM(A)=BM(A)-1:IF BM(A)>=0 THEN K=K+1:GOTO 790
676 BM(A)=DF:OP=ABS(BP(A))
680 DH=SGN(HC-BC(A)):DV=WD*SGN(HR-BR(A))
690 NP=OP+DV+DH:X=PEEK(NP):IF X=Y OR X=G THEN 760
700 NP=OP+DH:X=PEEK(NP):IF X=Y OR X=G THEN 760
710 NP=OP+DV:X=PEEK(NP):IF X=Y OR X=G THEN 760

```

```

720 C=0:NP=-OP:FOR D=1 TO 9
730 T=BP(A)+WD*DR(D)+DC(D)
740 X=PEEK(T):IF X=G OR X=Y THEN C=C+1:IF RND(1)>(C-1)/C THEN NP=T
750 NEXT D
760 BP(A)=NP:IF NP<0 THEN 790
762 POKE NP+DI,C1:POKE OP+DI,C3
765 POKE OP,G:POKE NP,M:K=K+1:NP=NP-UL:BR(A)=INT(NP/WD):
BC(A)=NP-WD*BR(A)
770 FOR X=0 TO B:IF BP(X)>0 THEN 780
775 IF ABS(BC(X)-BC(A))<2 AND ABS(BR(X)-BR(A))<2 THEN
    BP(X)=ABS(BP(X)):K=K+1
780 NEXT X
790 NEXT A:IF K=0 THEN 880
800 IF PEEK(HP)=M THEN 820
810 GOTO 540
820 PRINT "{home down rvs-on}
    YOU HAVE BEEN EATEN BY A WILD BEAST!!!"
830 BS=BS+1:X=M:POKE S1,30:POKE VL,15
840 FOR A=1 TO 27:POKE HP,X:FOR C=1 TO 20:NEXT:POKE S1,45:POKE HP,Y
850 POKE HP+DI,C2:FOR C=1 TO 40:NEXT:POKE S1,40:NEXT:POKE HP+DI,C3
860 POKE HP,93:FOR A=1 TO 5:POKE S1,170-A↑2:NEXT:POKE HP-WD,91
865 FOR A=1 TO 850:NEXT
870 GOTO 920
880 PRINT "{home down rvs-on}
    YOU'VE WON! THE WILD BEASTS ARE CAUGHT!!":YS=YS+1:X=83
890 POKE S1,30:POKE VL,15
900 FOR A=1 TO 30:POKE HP,X:FOR C=1 TO 10:NEXT C:POKE S1,255
910 POKE HP,Y:POKE HP+DI,C2:FOR C=1 TO 30:NEXT:POKE S1,235:NEXT
915 FOR A=1 TO 200:NEXT
920 POKE VL,0
930 PRINT "{clr 3°down}"
960 GOSUB 9100:IF YN THEN 290
980 PRINT "{clr 3°down wht}YOU PLAYED A TOTAL OF" BS+YS
    "ROUNDS.{down}"
990 A$="YOU":B$="THE WILD BEASTS"
1000 T=YS:GOSUB 2000:T=BS:GOSUB 2000
1010 C=INT(100*YS/(YS+BS)):A$=STR$(C)+"%":IF C=0 THEN A$="NONE"
1020 PRINT "YOU WON" A$ " OF THE TIME."
1030 ON INT((C+YS+BS)/20) GOTO 1050,1060,1070,1080,1080,1090,1090,
    1090,1090
1040 A$="PURINA(TM) BEAST CHOW":GOTO 1100
1050 A$="SURVIVOR":GOTO 1100
1060 A$="BEGINNING BEAST TRAPPER":GOTO 1100
1070 A$="ADVANCED BEAST TRAPPER":GOTO 1100
1080 A$="PROFESSIONAL BEAST TRAPPER":GOTO 1100
1090 A$="EXPERT PROFESSIONAL":GOTO 1100
1100 PRINT "{down}RATING:" A$ "{6°down}":END
1110 PRINT "{home rvs-off 14°right}SCORE: BEASTS:" BS " YOU:" YS

```

```

1120 RETURN
1130 VR=INT(RND(1)*21):VC=INT(RND(1)*40)
1140 VP=VR*WD+VC+UL:IF PEEK(VP)<>G THEN 1130
1150 RETURN
2000 PRINT A$;:IF T=0 THEN PRINT ",NEVER";
2010 PRINT ",CAUGHT,";B$;
2020 IF T>0 THEN PRINT T;"TIME";:IF T>1 THEN PRINT "S";
2030 PRINT ".":PRINT
2040 T$=A$:A$=B$:B$=T$:RETURN
2100 PRINT "{home 15°right}";
2110 PRINT ABS(BR(A)-HR);"{left}";ABS(BC(A)-HC);"{left}"
2120 IF PEEK(152)=0 THEN 2120
2130 RETURN
7000 PRINT "{clr wht 2°down}PICK,A,SKILL,LEVEL:{down cyn}":
7010 FOR I=1 TO 9:PRINT I,:NEXT I:PRINT:G=1:T=1:TM=0
7020 X=G:IF TI>TM THEN PRINT TAB(3*G-3);MID$("wht cyn",T,1);
  "{up}";G:T=3-T:TM=TI+10
7030 IF FNJ(0)=4 THEN IF G>1 THEN G=G-1
7040 IF FNJ(0)=6 THEN IF G<9 THEN G=G+1
7050 IF X<>G THEN PRINT TAB(3*X-3);"{cyn up}";X:T=1:TM=0
7060 IF FNB(0)=0 THEN 7020
7070 RETURN
9100 NO=0
9110 TM=0:SW=1
9115 PRINT "{wht}PLAY AGAIN?,{cyn}YES,{up}"
9120 IF TI<TM THEN 9150
9130 PRINT TAB(12+NO*4);MID$("wht cyn",SW,1);
  MID$("YESNO{shift-space}",3*NO+1,3);"{up}"
9140 SW=3-SW:TM=TI+15
9150 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
9160 IF JB AND JD=15 THEN YN=(NO=0):RETURN
9170 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 9115
9180 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 9115
9190 GOTO 9120
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$(".{+}",ZC,1);"{left}";:ZC=3-ZC:
  ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT ",{left}";:
  GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
  GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
  GOTO 60000

```

```
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$(" {21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$(" {21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:$:PRINT "{7°down cyn 9°space}"
(C)^1983^THE^CODE^WORKS"
62070 GET IN$:$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$(" {pur cyn}",K+1)LEFT$(MS$,Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

DIVE (Uses joystick)

By Glen Fisher



You command a crew of five divers who are trying to recover ancient silver treasure from a sunken ship at the bottom of the ocean. There is one problem: a ball of energy floats by periodically. Any contact with it is deadly. You control the divers with the joystick as they dive for the silver. Each diver must retrieve the treasure by touching it with his head and then resurfacing in exactly the same spot where he dove into the water. As the diver swims down to the treasure, his air tanks are gradually exhausted, as shown by the scale at the top of the screen. As you might expect, a diver who runs out of

air turns blue and dies. (Watching a diver after he's been hit by the energy ball isn't a pretty sight, either.)

If you choose the more difficult level of play, the divers are given less air, while at the easy level they get more air. The remaining divers in your crew are shown on the deck of the ship. The game is over when either all the treasure has been recovered or when there are no more divers remaining. Although you have a crew of five, it is considered very poor form to sacrifice the entire crew just to reward your greed. Your score, out of a possible 500, is listed at the end of the game.

```
0 PG$="^D^I^V^E^":AU$="BY^GLEN^FISHER":JF=1
1 REM DIVE -- BY GLEN FISHER
4 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 9/6/83 GF
90 GOTO 62000
100 DIM JD(15):FOR I=0 TO 15:READ JD(I):NEXT:QI=254:POKE VIC+32,6:
    POKE VIC+33,0
110 DATA -1,-1,-1,-1,-1,-1,1,-1,-1,-1,3,-1,2,0,-1
120 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
    16)=0
130 FOR I=0 TO 28:POKE SID+I,0:NEXT:POKE SID+6,240:POKE SID+4,17
200 FOR I=0 TO 3:READ RC(I),CC(I),RO(I),CO(I):NEXT
210 DATA -4,0,11,15, 0,4,15,4, 4,0,4,0, 0,-4,0,11
220 FOR I=1 TO 6:READ SS(I):NEXT
230 DATA TERRIBLE, POOR, FAIR, GOOD, "VERY^GOOD", EXCELLENT
240 FOR I=1 TO 6:READ FB(I):NEXT I
250 DATA 2,8,7,1,7,8
300 BR=CRT+24*WD
500 GOSUB 9000:GOSUB 8000
1100 PRINT "{clr}";CHR$(142);"{down wht}"
1105 PRINT "{7^space M}"
1110 PRINT "^_{4^@ I rvs-on V}_{rvs-off I 3^@}"
1120 PRINT "^_{* rvs-on}_{rvs-off 6^space rvs-on}_{rvs-off £}"
1130 PRINT "{blu 12^T cyn T blu 27^T wht}"
1200 ND=6:DD=-1:FOR I=2 TO 6:POKE SP+I,S0
1210 POKE VIC+39+I,10
```

```

1220 POKE VIC+I+I,C0+59-8*I:POKE VIC+I+I+1,R0+10
1230 NEXT I
1300 PRINT "{home 23°down}"
1310 FOR I=1 TO 6
1320 T=INT(RND(1)*(WD-2))+1
1330 A=BR+T:IF PEEK(A)<>32 THEN 1320
1340 PRINT TAB(T);"{Q up}"
1350 NEXT I:NP=6:NS=0
1400 LD=C0+87:RD=C0+93:TD=R0+33:BD=R0+40
1410 C1=C0+1:C9=C0+298:R1=BD+1:R9=R0+173
1420 MA=1+2*((C9-(LD+RD)/2)/CC(1)+(R9-TD)/RC(2))
1500 ND=ND-1:DD=DD+1:IF ND<1 OR NP=0 THEN 5000
1505 S=7-ND:POKE VIC+S+S,0:POKE VIC+S+S+1,0
1510 C=(LD+RD)/2:R=TD:POKE SP+2,S0
1520 IF S>2 THEN POKE VIC+S+S,C:POKE VIC+S+S+1,R
1530 POKE VIC+4,C:POKE VIC+5,R
1540 IF S>2 THEN POKE VIC+S+S+1,0
1550 POKE VIC+21,254:FT=INT(MA*10/SK+1)
1560 SF=FT/140
1570 PRINT "{home}AIR:{rvs-on 35°space}"
1600 DR=0:HB=0:HP=0:AR=FT:T=PEEK(VIC+30)+PEEK(VIC+31)
2000 D=FNJ(0):IF D>=0 THEN DR=D
2010 GET T$:IF T$="Q" THEN POKE VIC+21,0:GOTO 5200
2100 NC=C+CC(DR):NR=R+RC(DR)
2110 IF NC>=C1 AND NC<=C9 AND NR>=R1 AND NR<=R9 THEN 2200
2120 IF NC>=LD AND NC<=RD AND NR>=TD AND NR<=BD THEN 2200
2130 NR=R:NC=C:IF R>=R1 OR HP=0 THEN 2200
2140 POKE VIC+21,254:PRINT "{home 4°down}";TAB(4+NS);"{Q}"
2150 HP=0:NS=NS+1:IF NP=0 THEN 5000
2160 W=INT((AR+SF-1)/SF):T=INT(W/4):Z=W-4*T:AR=FT
2170 PRINT "{home rvs-on}";TAB(T+4);MID$("rvs-off H rvs-off K
    rvs-on N rvs-on",Z+Z+1,2)
2180 Z=Z+1:IF Z>3 THEN Z=0:T=T+1:IF T>34 THEN 2200
2190 GOTO 2170
2200 IF R<R1 THEN 2300
2210 AR=AR-1:W=INT((AR+SF-1)/SF)
2220 T=INT(W/4):Z=W-4*T
2240 PRINT "{home rvs-on}";TAB(T+4);MID$("rvs-off").{rvs-off H
    rvs-off K rvs-on N",Z+Z+1,2)
2250 IF AR<=0 THEN POKE VIC+41,6:FOR I=1 TO 500:NEXT I:GOTO 4300
2300 C=NC:R=NR:OD=DR:HB=HB AND NOT 5
2310 CL=C AND 255:HB=HB AND NOT 133:IF C>255 THEN HB=HB OR 132
2330 PR=R+RO(DR):PC=C+CO(DR):PL=PC AND 255:IF PC>255 THEN HB=HB OR 1
2340 POKE SP+2,S0+DR
2350 POKE VIC+4,CL:POKE VIC+5,R:POKE VIC+16,HB
2360 POKE VIC,PL:POKE VIC+1,PR
2370 GOTO 2600
2500 IF DR=OD THEN 3000

```

2510 HB=HB AND NOT 1
2520 P=C+CO(DR):IF P>255 THEN HB=HB OR 1
2530 POKE SP+2,S0+DR:POKE VIC+16,HB
2540 POKE VIC,P AND 255:POKE VIC+1,R+RO(DR)
2550 OD=DR
2600 IF HP OR R<R9 THEN 3000
2610 T=PEEK(VIC+31):T=TI+1
2620 T=PEEK(VIC+31):IF (T AND 4)=0 THEN 3000
2630 A=INT((C-C0)/8)+BR+1
2640 POKE A,32
2650 POKE VIC+21,255:HP=1:NP=NP-1
2660 GOTO 3000
3000 IF MS THEN 3100
3010 MR=INT(RND(1)*(R9-R1))+R1
3020 MC=INT(RND(1)*(C0-12))
3030 POKE VIC+3,0
3040 MS=INT(RND(1)*CC(1)*2)
3045 HB=HB AND NOT 2:POKE VIC+16,HB
3050 POKE VIC+2,MC:POKE VIC+3,MR
3060 T=PEEK(VIC+30):GOTO 2000
3100 MC=MC+MS:Z=PEEK(VIC+30)
3110 IF MC>255 THEN HB=HB OR 2:POKE VIC+16,HB
3120 POKE VIC+2,MC AND 255
3130 IF MC>C9+20 THEN MS=0
3140 Z=Z OR PEEK(VIC+30)
3145 FB=FB+1:IF FB>6 THEN FB=1
3150 POKE VIC+40,FB(FB)
3155 IF (Z AND 2)=0 THEN 2000
3160 GOTO 4000
4000 POKE VIC+46,2:MS=0
4010 POKE VIC+14,PEEK(VIC+4):POKE VIC+15,PEEK(VIC+5)
4020 GOSUB 4900:POKE VIC+46,8:POKE VIC+3,0
4030 GOSUB 4900:POKE VIC+46,7
4040 GOSUB 4900:POKE VIC+46,1
4050 GOSUB 4900:POKE VIC+46,7:POKE VIC+41,1
4060 GOSUB 4900:POKE VIC+46,8:POKE VIC+41,7
4070 GOSUB 4900:POKE VIC+15,0:POKE VIC+41,8
4080 GOSUB 4900:POKE VIC+41,11
4090 GOSUB 4900:GOTO 4300
4300 POKE VIC+27,5:P=PEEK(VIC+1):IF DR<>3 THEN POKE SP+2,S0+1
4310 R=R+1:P=P+1
4320 IF R<256 THEN POKE VIC+5,R
4330 IF P<256 THEN POKE VIC+1,P
4340 IF R<256 OR P<256 THEN 4310
4350 POKE VIC+41,10:POKE VIC+27,0
4360 GOTO 1500
4900 T=TI+4
4910 IF TI<T THEN 4910
4920 RETURN

```

5000 T=TI+60
5010 IF TI<T THEN 5010
5020 POKE VIC+21,0
5030 PRINT "{clr}";CHR$(14);"{2°down Y}OU HAVE SALVAGED";:
  IF NS=0 THEN PRINT "~NO~";
5035 IF NS>0 THEN PRINT NS;
5040 PRINT "TREASURE";:IF NS<>1 THEN PRINT "S";
5050 PRINT "~OUT_OF":
  PRINT "{down}A POSSIBLE SIX, AND HAVE SACRIFICED"
5060 PRINT "{down}THE~";:IF DD=1 THEN PRINT "LIFE_OF";:GOTO 5080
5070 PRINT "LIVES_OF";
5080 PRINT DD;"DIVER";:IF DD<>1 THEN PRINT "S";
5090 P=100*NS-4*DD:PRINT ".~{T}HIS~GIVES"
5100 PRINT "{down}YOU AN OVERALL SCORE_OF";:IF P<0 THEN P=0
5110 PRINT P;"{left}, WHICH IS"
5120 P=INT((P+60)/132)+1
5130 PRINT "{down 6°space}";S$(P);"{2°down}"
5140 GOSUB 8100
5150 IF A THEN 1100
5200 PRINT "{clr blu}";CHR$(142):POKE VIC+32,14:POKE VIC+33,6:END
7999 END
8000 PRINT "{clr wht}PICK A SKILL LEVEL:{down cyn}":
8010 FOR I=1 TO 10:PRINT I;:NEXT I:PRINT:SK=5:T=1:TM=0
8020 X=SK:IF TI>TM THEN PRINT TAB(3*SK-3);MID$("{wht cyn}",T,1);
  "{up}";SK:T=3-T:TM=TI+5
8030 IF FNJ(0)=3 THEN IF SK>1 THEN SK=SK-1
8040 IF FNJ(0)=1 THEN IF SK<10 THEN SK=SK+1
8050 IF X<>SK THEN PRINT TAB(3*X-3);"{cyn up}";X:T=1:TM=0
8060 IF FNB(0)=0 THEN 8020
8070 RETURN
8100 PRINT "{down wht}PLAY AGAIN?~{shift-space cyn}NO~YES":
8110 A=1:T=1:TM=0
8120 X=A:IF TI<TM THEN 8150
8130 PRINT MID$("{wht cyn}",T,1);TAB(13+4*A);"{up}";MID$("NO~YES",
  3*A+1,3)
8140 T=3-T:TM=TI+10
8150 IF FNJ(0)=3 THEN A=0
8160 IF FNJ(0)=1 THEN A=1
8170 IF X<>A THEN PRINT TAB(13+4*X);"{up cyn}";MID$("NO~YES",3*X+1,
  3):T=1:TM=0
8190 IF FNB(0)=0 THEN 8120
8200 RETURN
9000 SB=16384-512:S0=(SB/64) AND 255:SP=CRT+1016
9010 FOR I=0 TO 6:A=SB+64*I:READ N
9015 PRINT "{home down}";7-I
9020 FOR J=0 TO N-1:READ T:POKE A+J,T:NEXT
9030 FOR J=N TO 63:POKE A+J,0:NEXT
9040 NEXT I

```

```

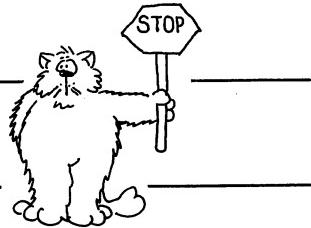
9050 R0=50:C0=24:FOR I=1 TO 15 STEP 2:POKE VIC+I,0:NEXT:
    POKE VIC+16,0
9060 POKE SP,S0+4:POKE SP+1,S0+5:POKE SP+7,S0+6
9070 POKE VIC+40,7:POKE VIC+37,7
9080 POKE VIC+28,2
9090 POKE VIC+21,255:RETURN
40000 DATA 62,0,112,0,0,248,0,0,248,0,0,248,0,0,112,0,3,254,0,7,255,
    0,15,255
40001 DATA 128,29,253,192,57,252,224,49,252,96,33,84,32,1,84,0,1,84,
    0,1,84,0,1
40002 DATA 220,0,1,140,0,1,140,0,1,140,0,3,142,0,3,6
40003 DATA 56,0,0,0,0,0,0,112,0,0,56,0,0,28,0,0,14,0,192,7,0,254,
    191,0,124
40004 DATA 191,112,2,191,248,2,191,248,2,191,248,124,191,112,254,191,
    0,192,7,0
40005 DATA 0,14,0,0,28,0,0,56,0,0,112
40006 DATA 62,3,6,0,3,142,0,1,140,0,1,140,0,1,140,0,1,140,0,1,220,0,
    1,84,0,1,84
40007 DATA 0,33,84,32,49,84,96,57,252,224,29,253,192,15,255,128,7,
    255,0,3,254,0
40008 DATA 0,112,0,0,248,0,0,248,0,0,248,0,0,112
40009 DATA 56,0,0,0,0,0,0,112,0,0,224,0,1,192,0,3,128,0,7,0,24,7,
    235,248,119
40010 DATA 233,240,255,234,0,255,234,0,255,234,0,119,233,240,7,235,
    248,7,0,24,3
40011 DATA 128,0,1,192,0,0,224,0,0,112
40012 DATA 16,120,0,0,252,0,0,252,0,0,252,0,0,252,0,0,120
40013 DATA 32,10,128,0,41,160,0,37,96,0,165,104,0,149,88,0,149,88,0,
    149,88,0
40014 DATA 165,104,0,37,96,0,41,160,0,10,128
40015 DATA 62,1,252,0,7,255,0,31,255,192,63,255,224,63,255,224,127,
    255,240,127
40016 DATA 255,240,255,255,248,255,255,248,255,255,248,255,255,248,
    255,255,248
40017 DATA 255,255,248,255,255,248,127,255,240,127,255,240,63,255,
    224,63,255
40018 DATA 224,31,255,192,7,255,0,1,252
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MIDS("^.{+}",ZC,1); "{left}"; :ZC=3-ZC:
    ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT ".{left}"; :
    GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MIDS(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}"; :
    GOTO 60010

```

```
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:  
      GOTO 60000  
60140 GOTO 60010  
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:  
      CM=55296  
61010 QL=214:QI=254:RETURN  
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):  
      TB=(40-LN)/2  
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$(" {21°space}",LN)  
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"  
      TAB(TB)LEFT$(" {21°space}",LN)  
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$  
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN  
      MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"  
62050 TB=((40-LEN(MS$))/2)  
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}  
      (C)^1983^THE^CODE^WORKS"  
62070 GET IN$:IF IN$<>"" THEN 62120  
62080 IF (PEEK(JS) AND 16)=0 THEN 62120  
62090 PRINT "{home 14°down}" TAB(TB)LEFT$(" {pur cyn}",K+1)LEFT$(MS$,  
      Q)  
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K  
62110 GOTO 62070  
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```

STOP

By Randall Lockwood



STOP is a game for two players. (You may play against the C-64 or against another person.) Two players take turns rolling the dice and moving their pieces up the board. The board resembles a diamond and is made up of 11 columns that are numbered from 2 to 12. The object of the game is to capture three of the 11 columns on the game board by steadily advancing your pieces up the columns. A column is captured by ending your turn with your piece at the top of a column.

In each turn, your piece to be moved is represented by a temporary marker showing where your piece will move at the end of the turn. (Your piece doesn't actually move until your turn is over. The temporary markers do all the moving within a turn.) Your temporary marker begins moving from where your piece is in the column. If you don't have a piece in a column, the temporary marker will start at the bottom of the column. You can't have more than three temporary markers on the board during a turn. As long as you can keep moving the markers, you can keep rolling the dice.

When it's your turn to play, the C-64 will roll the four dice and then divide them into all possible pairs. For example, the roll 1-2-4-6 will be divided into the pairs "3-10," "5-8," "8-5," and "10-3." You choose one of the pairs to move your temporary markers up the column. For each number in the pairing, the C-64 will move the marker in the corresponding column up one position. For example, if you have a temporary marker in column 3 and you choose "3-10," then your marker in column 3 would move up one position. If you also have a marker in column 10, then it too would move up one position. If there's no marker in the column, one will be added above the piece, or at the bottom if there isn't a piece in that column.

Why list the pairs twice, such as "3-10" and "10-3"

in the preceding example? The reason is that sometimes the order of the numbers makes a difference. Suppose both the numbers in the pairing were for columns you do not already have a marker in, and two of the markers are already on the board. You would then have to make a choice between one of the two numbers in the pair because both numbers require a new marker. The first number in the pair you choose is where the last marker will be placed, and the second number is ignored. The pairs are listed twice, in opposite order, so you can choose which number you want to use. You may continue to roll as often as you like, as long as you don't get "zapped."

If you get a roll which won't let you move or add a marker, then you've been "zapped." Your turn is ended, and worse, you lose all the movement you achieved during the turn. The markers are removed, and your pieces stay where they were at the start of the turn.

If you choose to stop rolling, the temporary markers are replaced by the pieces from the same columns. (New pieces are added to replace markers in columns that are without pieces. There's no limit to how many pieces you can have on the board at once.) If any one of the newly moved pieces is at the top of its column, you've captured that column. The other player's piece in that column (if any) is removed. Capture three columns, and you've won the game.

You might think that the short columns are the easiest to capture, but you would be wrong. While the 2 and 12 columns are the shortest, 2 and 12 are also the least likely numbers for you to roll. The most likely number is a 7, but the 7 column is the longest one on the board. The net result is that all the columns are equally difficult to capture.

```
0 PG$={shift-space}S.T.O.P.:AU$="BY RANDALL LOCKWOOD":JF=0
1 REM STOP -- BY RANDALL LOCKWOOD
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
```

```

6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 9/6/83 GF
90 GOTO 62000
100 U$="{}home 20°down":B$="{}8°space"+CR$
101 DIM JD(15):FOR I=0 TO 15:READ JD(I):NEXT
102 DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,8,5
103 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
16)=0
104 FOR I=0 TO 28:POKE SID+I,0:NEXT:POKE SID+6,240:POKE SID+4,17
110 FQ=SID+1:VL=SID+24:QI=3:DI=CM-CRT:TN=15:SP=WD*2:C$="{}19°space"
120 DIM T(12),X1(12),P1(12),P(12),R(4),RR(6,1)
130 DIM PS%(2,12),GC(2),W%(12),X(12),PF(2,12),PN$(2),PS$(2)
140 PRINT "{}clr down}WANT TO PLAY AGAINST THE 64? ";
150 GOSUB 60000:PRINT:PET=(LEFT$(IN$,1)<>"N")
160 IF PET THEN PN$(1)="YOU":PN$(2)="I":PS$(1)="YOUR":PS$(2)="MY":
GOTO 210
170 QI=20:FOR I=1 TO 2:PRINT "PLAYER";I;"{}left} 'S NAME? ";
GOSUB 60000
180 IF IN$="" THEN IN$="PLAYER "+CHR$(I+48)
190 PN$(I)=LEFT$(IN$,8):PS$(I)=PN$(I)+"'S"
200 NEXT I:QI=3
210 GC(1)=81:GC(2)=90:CL(1)=10:CL(2)=14:BK=11
215 FOR I=2 TO 12:T(I)=12-2*ABS(7-I):NEXT
230 FOR I=2 TO 12:P(I)=CRT+3*I-2+(14+T(I))*WD:NEXT
240 Z$="{}cyn 2°@}~{}pur}":PRINT "{}clr pur}":PRINT "{}up}";TAB(19),
"~7{down}"
250 PRINT TAB(16) "~6~" Z$ "~8{down}"
260 PRINT TAB(13); "~5~" Z$Z$Z$ "~9{down}"
270 PRINT TAB(10); "~4~" Z$Z$Z$Z$Z$Z$ "10{down}"
280 PRINT TAB(7); "~3~" Z$Z$Z$Z$Z$Z$Z$Z$ "11{down}"
290 PRINT "{}5°space}2~" Z$Z$Z$Z$Z$Z$Z$Z$Z$Z$ "12{down}"
300 PRINT "{}4°space}" Z$Z$Z$Z$Z$Z$Z$Z$Z$Z$Z$ " {down}"
310 PRINT "{}4°space}" Z$Z$Z$Z$Z$Z$Z$Z$Z$Z$Z$Z$ " {down}"
320 PRINT TAB(7);Z$Z$Z$Z$Z$Z$Z$Z$Z$Z$ " {down}"
330 PRINT TAB(10);Z$Z$Z$Z$Z$Z$Z$ " {down}"
340 PRINT TAB(13)Z$Z$Z$Z$Z$ " {down}"
350 PRINT TAB(16)Z$Z$Z$ " {down}"
360 PRINT TAB(19);Z$ " {home}"
400 FOR I=1 TO 2:FOR J=2 TO 12:PS%(I,J)=0:PF(I,J)=0:NEXT J:CW(I)=0:
NEXT I
410 FOR J=2 TO 12:W%(J)=0:NEXT J
470 PL=2
475 PRINT "{}home}";TAB(31); "THE,DICE"
480 C=3
490 PL=PL+1:IF PL>2 THEN PL=1
510 FOR I=2 TO 12:X(I)=0:NEXT:FR=0
515 PRINT "{}home}";C$
```

```

520 PRINT "{home}" LEFT$("{blk cyn}",PL);PS$(PL);"~TURN~~";
CHR$(GC(PL)+128)
540 IF PL=2 AND PET THEN 1200
545 GOSUB 2200:IF R$="Q" THEN PRINT "{clr}":END
546 IF R$="R" THEN 870
580 PRINT U$;"{rvs-on}~STOP~":T=TI+300
590 FOR J=2 TO 12
600 IF X(J)=0 THEN 720
605 SD=135/2↑(PF(PL,J)/12):GOSUB 2600
610 IF PF(PL,J)<=T(J) THEN 680
620 CW(PL)=CW(PL)+1:W%(J)=1
630 FOR K=1 TO T(J)+1:Z=P(J)-SP*K
635 POKE Z,GC(PL):POKE Z+1,GC(PL)
640 POKE Z+DI,CL(PL):POKE Z+DI+1,CL(PL)
650 NEXT K
670 GOTO 720
680 IF PS%(PL,J)>0 THEN L=P(J)-SP*PS%(PL,J)+PL-1:POKE L,100:
POKE L+DI,BK
690 L=P(J)-SP*PF(PL,J)+PL-1:POKE L,GC(PL)
695 POKE L+DI,CL(PL)
710 PS%(PL,J)=PF(PL,J)
720 GOSUB 2650:NEXT J
723 IF TI<T THEN 723
725 PRINT U$;B$
730 IF CW(PL)>=3 THEN 1100
740 GOTO 480
870 PRINT U$;"{rvs-on}~ZAP!!~":SD=255:TN=25
875 GOSUB 2600:FOR Z=1 TO 750:NEXT
880 GOSUB 2650:FOR Z=1 TO 750:NEXT
890 PRINT U$;"{7°space}":FOR J=2 TO 12
900 IF X(J)=0 THEN 960
910 IF PF(PL,J)>T(J) THEN 930
920 L=P(J)-SP*PF(PL,J)+PL-1:POKE L,100:POKE L+DI,BK:GOTO 950
930 PRINT LEFT$(U$,13-T(J));TAB(3*T(J-2));RIGHT$(STR$(J),2)
950 PF(PL,J)=PS%(PL,J)
960 NEXT J:GOTO 480
1000 GOSUB 2490
1020 T=M(1):GOSUB 1900
1030 T=M(2):GOSUB 1900
1040 IF PL=2 AND PET THEN FOR Z=1 TO 2000:NEXT
1050 GOTO 540
1100 Z$="S":IF PET THEN Z$=""
1105 RESTORE:I=1:Z=18:T=TI
1110 PRINT "{home}";C$:FOR K=1 TO 30
1115 PRINT "{home}" CHR$(Z);PN$(PL);".WIN";Z$;T=TI+6
1120 IF K=I THEN READ SD,I:I=K+I:GOSUB 2650:GOSUB 2600
1125 IF TI<T THEN 1125
1130 Z=(Z+128) AND 255
1140 NEXT K

```

```

1145 GOSUB 2650
1150 PRINT U$;"AGAIN?":GOSUB 60000
1160 IF LEFT$(IN$,1)="N" THEN END
1170 GOTO 240
1180 DATA 214,4,214,2,214,6,202,2,214,2,255,2,241,2,214,0
1200 IF C>0 THEN 1230
1210 J=0:FOR I=2 TO 12:IF X(I) THEN J=J+I:IF PF(PL,I)>T(I) THEN 580
1215 NEXT:J=J/3
1217 IF RND(1)*(7-ABS(7-J))>FR THEN 1230
1220 GOTO 580
1230 GOSUB 2300:PRINT U$ "{rvs-on}THINKING"
1240 TS=0:FOR I=1 TO NP
1250 R1=RR(I,0):R2=RR(I,1)
1260 GOSUB 1400
1290 NEXT I
1300 PRINT U$;B$
1310 IF TS=0 THEN 870
1330 GOTO 1000
1400 FOR Z=2 TO 12:X1(Z)=X(Z):P1(Z)=PS%(PL,Z):NEXT Z:C1=C
1410 SC=0:T=R1:GOSUB 1500:T=R2:GOSUB 1500
1420 IF SC>TS THEN TS=SC:M(1)=R1:M(2)=R2:M=I
1430 RETURN
1500 IF X1(T)=0 AND C1=0 THEN RETURN
1501 IF W%(T) OR P1(T)>T(T) THEN RETURN
1505 SC=SC+1
1520 IF X1(T) THEN SC=SC+2
1530 IF P1(T)=T(T) THEN SC=SC+5
1540 IF X1(T)=0 THEN C1=C1-1
1550 P1(T)=P1(T)+1:X1(T)=1:RETURN
1800 PRINT U$;LEFT$("{red blu}",PL); "WHICH PAIR?":QI=3:GOSUB 60000
1810 PRINT U$; "{11°space}":PRINT B$
1815 IF LEFT$(IN$,1)="Q" THEN V=0:R$="Q":RETURN
1820 IF IN$="" AND NP=1 THEN IN$="1"
1825 IF IN$="" THEN 1800
1830 M=VAL(IN$):IF IN$="0" THEN RETURN
1835 IF M<1 OR M>NP THEN 1800
1837 GOSUB 2490
1840 V=0:T=RR(M,0):GOSUB 1900
1850 T=RR(M,1):GOSUB 1900
1860 RETURN
1900 IF W%(T) THEN RETURN
1910 I=PF(PL,T):IF I<>PS%(PL,T) THEN 1950
1920 IF C>0 THEN GOSUB 2000:X(T)=1:C=C-1
1930 RETURN
1950 IF I<=T(T) THEN GOSUB 2000
1960 RETURN
2000 SD=255/2↑(PF(PL,T)/12):GOSUB 2600
2005 IF PF(PL,T)=PS%(PL,T) THEN 2020

```

```

2010 L=P(T)-SP*PF(PL,T)+PL-1:POKE L,100:POKE L+DI,BK
2020 PF(PL,T)=PF(PL,T)+1
2030 L=P(T)-SP*PF(PL,T)+PL-1:POKE L,214:POKE L+DI,CL(PL)
2040 V=1:FR=FR+1:GOSUB 2650:RETURN
2200 IF C=3 THEN 2235
2205 PRINT U$;"{rvs-on}R{rvs-off up left @ down}^OR^{rvs-on}
S{rvs-off up left @ down}?{4°space}"
2210 QI=3:GOSUB 60000:RS=LEFT$(IN$,1)
2215 PRINT "{3°up}";B$;B$;B$;
2220 IF RS="S" OR RS="Q" THEN RETURN
2230 IF RS<>"" AND RS<>"R" THEN 2205
2235 RS="R":GOSUB 2300:GOSUB 2700:IF ZP THEN RETURN
2240 V=0:GOSUB 1800:IF V THEN 2205
2250 RETURN
2300 PRINT "{home down}":PRINT TAB(30);B$:PRINT "{down}";TAB(30);B$
2305 PRINT "{home down}":FOR I=1 TO 6:PRINT B$;:NEXT
2310 FOR I=1 TO 4:R(I)=INT(6*RND(1)+1):NEXT
2320 PRINT "{home down wht}":PRINT TAB(33);"{rvs-on}";CHR$(R(1)+48);
"2°right>";CHR$(R(2)+48)
2330 PRINT "2°down" TAB(33);"{rvs-on}";CHR$(R(3)+48);"2°right";
CHR$(R(4)+48)
2340 T=R(1)+R(2)+R(3)+R(4)
2350 NP=0:FOR I=1 TO 3:FOR J=I+1 TO 4
2360 R1=R(I)+R(J):R2=T-R1
2370 IF NP=0 THEN 2410
2380 FOR M=1 TO NP
2390 IF RR(M,0)=R1 AND RR(M,1)=R2 THEN 2420
2400 NEXT M
2410 NP=NP+1:RR(NP,0)=R1:RR(NP,1)=R2
2420 NEXT J,I
2430 PRINT "{home down}":FOR M=1 TO NP:GOSUB 2500:NEXT M
2440 RETURN
2490 PRINT LEFT$("home 7°down",M+2);"{rvs-on}";
2500 PRINT MID$(STR$(M),2);":";
2510 PRINT RIGHT$(STR$(RR(M,0)),2);".";
2520 PRINT RIGHT$(STR$(RR(M,1)),2)
2530 RETURN
2600 POKE VL,15:POKE FQ,SD:TN=15:RETURN
2650 POKE VL,0:RETURN
2700 ZP=0:IF C>0 THEN RETURN
2710 FOR Z=1 TO NP
2720 IF X(RR(Z,0)) OR X(RR(Z,1)) THEN RETURN
2730 NEXT Z:ZP=1:RETURN
2995 RETURN
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("_.+",ZC,1);"{left}";:ZC=3-ZC:
ZT=TI+15

```

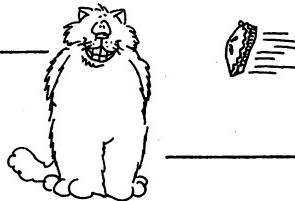
```

60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT ".{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
(C)_1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>" " THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

PIEGRAM (Uses joystick)

By Glen Fisher



You are about to indulge in the classic prank of pie throwing. You and the computer take turns trying to hit each other with pies; if the pies get too close for comfort, you and the computer can dodge to a different square on the board.

The playing board is a 10×10 grid. When the game starts, you can hide in any of the squares. Your position will be marked by a green "X." The computer is also in one of the squares, but you can't see it and it can't see you.

You are first given the choice of throwing a pie or moving. Use the joystick to make your selection, and then press the joystick button. If you choose to move, move the cursor to the square you want to move to and press the joystick button. You can move only one square per turn. Also, you can't move into a square a pie has landed in. If you try to move somewhere you're not allowed, the cursor will reappear, and you'll have to move somewhere else.

If you choose to throw a pie, move the cursor to the square where you want to throw it from and press the button. After each pie toss, you are given a clue to

help you find your opponent. The pie that lands in the selected square will be assigned the number one. On the side of the board, that same number will appear, and next to it will be another number telling how far away from the computer's square your pie landed. This number is computed by the Pythagorean Theorem. If you are good at using the Pythagorean Theorem and finding square roots, then you can deduce which squares the computer could be hiding in (if the computer hasn't moved). If you don't want to be quite that precise, you can estimate.

Meanwhile, the computer is trying to hit you with a pie or is trying to move away from your pie throw. The computer must play by the same rules you follow when it comes to throwing a pie or moving. But it has the advantage of being able to make quick and thorough calculations to determine your position if you do not occasionally move around.

The game is over when you hit the computer with a pie, or it throws a pie that hits you. Oh—don't forget to clean up the mess when you're done!

```
0 PG$="^P^E^G^R^A^M^":AU$="BY^GLEN^FISHER":JF=1
1 REM PIEGRAM -- BY GLEN FISHER
4 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
8 :
10 REM AS OF 9/6/83 GF
90 GOTO 62000
100 DIM B%(10,10),HS(4,3),CS(4,3),JD(15):QI=10
101 FOR I=0 TO 15:READ JD(I):NEXT:DATA 0,0,0,0,0,3,9,6,0,1,7,4,0,2,
    8,5
102 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
    16)=0
120 FOR I=1 TO 9:READ RC(I),CC(I):NEXT
130 DATA 1,-1,1,0,1,1,0,-1,0,0,0,1,-1,-1,-1,0,-1,1
160 BL$="{rvs-off 39°space}"
200 DN$="{home 10°down}"
500 FOR I=0 TO 4:FOR J=0 TO 3:HS(I,J)=0:CS(I,J)=0:NEXT J,I:HS=0:
    CS=0
```

```

510 FOR I=1 TO 10:FOR J=1 TO 10:B%(I,J)=0:NEXT J,I
600 GOSUB 9000:HR=0:HC=0:R=5:C=5
620 PRINT "{home blu}WHERE ARE YOU?":GOSUB 9100:IF C$="~" THEN 620
630 HR=R:HC=C:GOSUB 9300:GOSUB 9400:PRINT RV$:SY$
650 PR=INT(RND(1)*10)+1:PC=INT(RND(1)*10)+1
1000 PRINT "{home}";BL$:PRINT BL$ "{up}":GOSUB 8200
1020 IF IN$="Q" THEN 3120
1030 T$="MOV":IF YN=0 THEN T$="THROW"
1040 PRINT "{home}";BL$:PRINT "{home blu}WHERE ARE YOU~";T$;
    "ING~TO?"
1050 R=5:C=5:IF YN=1 THEN R=HR:C=HC
1060 SB=B%(HR,HC):B%(HR,HC)=2:GOSUB 9100:B%(HR,HC)=SB:IF C$="~" THEN
    1000
1070 IF YN=0 THEN 1200
1100 IF ABS(R-HR)>1 OR ABS(C-HC)>1 THEN 1050
1110 IF B%(R,C)<>0 THEN 1050
1120 SR=HR:SC=HC:HR=R:HC=C:R=SR:C=SC:GOSUB 9400:GOSUB 9300:
    PRINT RV$:SY$
1130 R=HR:C=HC:GOSUB 9400:GOSUB 9300:PRINT RV$:SY$
1140 FOR I=0 TO 2:CS(I,3)=CS(I,3)+5:NEXT I
1150 GOTO 2000
1200 IF B%(R,C)<>0 THEN 1050
1210 IF R=HR AND C=HC THEN 1050
1400 HS(3,0)=R:HS(3,1)=C:HS(3,2)=INT(100*SQR((R-PR)↑2+(C-PC)↑2))/100
1410 FOR I=0 TO 3:I1=I+1:R=HS(I,0):C=HS(I,1)
1420 IF R>0 THEN B%(R,C)=B%(R,C)+1:GOSUB 9300:GOSUB 9400:
    PRINT RV$:SY$
1430 :FOR J=0 TO 3:HS(I,J)=HS(I1,J):NEXT J
1440 NEXT I:PRINT "{home down}"
1450 FOR I=2 TO 0 STEP -1
1460 IF HS(I,0)>0 THEN PRINT 3-I;HS(I,2);"{left 4° space}"
1470 NEXT I
1480 IF HS(2,2)=0 THEN 3000
1490 HS=HS+1:GOTO 2000
2000 PRINT "{home}";BL$:PRINT DN$;"{7° space}"
2010 R=PR-HS(2,0):C=PC-HS(2,1):D=R*R+C*C
2020 IF D<=2 THEN 2100
2030 IF RND(1)>D/100 THEN 2500
2100 R=INT(RND(1)*10)+1:C=INT(RND(1)*10)+1
2120 R=PR+SGN(R-PR):C=PC+SGN(C-PC)
2130 IF B%(R,C)<>0 THEN 2100
2135 PC=C:PR=R
2140 PRINT DN$;"{red} I~MOVED"
2150 GOTO 1000
2500 IF CS>2 THEN 2600
2510 R=INT(RND(1)*10)+1:C=INT(RND(1)*10)+1
2520 IF B%(R,C)<>0 THEN 2510
2530 IF R=PR AND C=PC THEN 2510

```

```

2540 GOTO 2900
2600 MR=0:MC=0:MS=-1
2610 FOR I=1 TO 10:FOR J=1 TO 10:S=0
2620 IF B%(I,J)<>0 THEN 2850
2630 R=I:C=J:GOSUB 9300:PRINT "{pur rvs-on}?{rvs-off left}";
2640 FOR K=0 TO 2:R=CS(K,0):C=CS(K,1):D=CS(K,2):W=CS(K,3):L=D-W:
H=D+W
2650 I0=I-R:J0=J-C:T=I0*I0+J0*J0
2660 IF L<=T AND T<=H THEN S=S+1
2670 NEXT K
2700 IF S>MS THEN MR=I:MC=J:MS=S
2800 R=I:C=J:GOSUB 9400:PRINT RV$;SY$
2850 NEXT J,I
2860 R=MR:C=MC
2900 CS(3,0)=R:CS(3,1)=C:CS(3,2)=(R-HR)↑2+(C-HC)↑2:CS(3,3)=0
2910 FOR I=0 TO 3:I1=I+1:R=CS(I,0):C=CS(I,1)
2920 IF R>0 THEN B%(R,C)=B%(R,C)-1:GOSUB 9300:GOSUB 9400:
PRINT RV$;SY$
2930 FOR J=0 TO 3:CS(I,J)=CS(I1,J):NEXT J
2940 NEXT I:PRINT DN$;"{red}I~THREW":PRINT "{rvs-on 10°space}"
2950 FOR I=2 TO 0 STEP -1:IF CS(I,0)=0 THEN 2970
2960 PRINT "{rvs-on}";3-I;"{left 8°space 7°left}";INT(100*SQR(CS(I,
2)))/100
2970 NEXT I:PRINT "{rvs-on 10°space}"
2980 IF CS(2,2)=0 THEN 3000
2990 CS=CS+1:GOTO 1000
3000 W$="YOU":L$="ME":SH=0:IF B%(R,C)<0 THEN W$="I":L$="YOU":SH=1
3010 PRINT "{home}";BL$:PRINT "{home}";W$;"~GOT~";L$;"."
3020 GOSUB 9300:FOR I=1 TO 100:PRINT MID$("{rvs-on rvs-off}",(I AND
I)+1,1);"*{left}";
3025 FOR J=1 TO 10:NEXT J,I
3030 IF SH THEN R=PR:C=PC:GOSUB 9300:PRINT "{rvs-on V}":
PRINT DN$;"{rvs-on V rvs-off}~={_ME_}"
3100 PRINT "{home rvs-off}";:GOSUB 8100
3110 IF YN THEN 500
3120 PRINT "{clr}":END
8100 NO=0
8110 TM=0:SW=1
8115 PRINT "{home blu}PLAY~AGAIN?~{cyn}YES~NO"
8120 IF TI<TM THEN 8150
8130 PRINT "{up}";TAB(12+NO*4);MID$("{wht cyn}",SW,1);
MID$("YESNO{shift-space}",3*NO+1,3)
8140 SW=3-SW:TM=TI+15
8150 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
8160 IF JB AND JD=15 THEN YN=(NO=0):RETURN
8170 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 8115
8180 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 8115
8190 GOTO 8120

```

```

8200 NO=1:PRINT "{home}" BL$ "{home}";
8210 TM=0:SW=1
8215 PRINT "{home blu}MOVE,OR,THROW?"
8216 IF FNB(0) THEN 8216
8220 IF TI<TM THEN 8250
8230 PRINT "{home}" TAB(NO*8);MID$( "{wht blu}",SW,1);
    MID$("MOVE,THROW",5*NO+1,5)
8240 SW=3-SW:TM=TI+15
8250 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
8255 GET IN$:IF IN$="Q" THEN RETURN
8260 IF JB AND JD=15 THEN YN=(NO=0):RETURN
8270 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 8215
8280 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 8215
8290 GOTO 8220
9000 R$=" {Q*+*+*+*+*+*+*+*+*+*+*W} ":S$="- - - - - - - - - - - - "
9010 BP=18
9050 PRINT "{clr down}":PRINT TAB(BP); "{cyn A*R*R*R*R*R*R*R*S}":
    PRINT TAB(BP);S$
9060 FOR I=1 TO 9:PRINT TAB(BP);R$:PRINT TAB(BP);S$:NEXT I
9070 PRINT TAB(BP); "{z*E*E*E*E*E*E*E*E*X wht}"
9080 RETURN
9100 SW=1:TM=0
9105 IF FNB(0) THEN 9105
9110 PRINT "{home 2°down}":PRINT TAB(BP+1);
9140 GOSUB 9300:GOSUB 9400:GOTO 9170
9150 T=FNJ(0):IF T<>5 OR FNB(0) THEN 9200
9160 IF TI<TM THEN 9150
9170 PRINT MID$(" {rvs-on rvs-off}",SW,1); "{pur}";SY$;" {left
    rvs-off}";:TM=TI+15:SW=3-SW:GOTO 9150
9200 PRINT RV$;SY$;" {left}";:IF T=5 THEN RETURN
9220 R=R+RC(T):C=C+CC(T)
9230 IF R<1 THEN R=10
9240 IF R>10 THEN R=1
9250 IF C<1 THEN C=10
9260 IF C>10 THEN C=1
9270 SW=1:TM=0:GOTO 9140
9300 PRINT "{home}":POKE QL,2+2*R:PRINT "{up}";TAB(BP+2*C-1);:RETURN
9400 SY$=MID$(" .123*",1+ABS(B%(R,C)),1)
9410 IF R=HR AND C=HC THEN SY$="{V}"
9420 RV$=MID$(" {red 2°cyn}",2+SGN(B%(R,C)),1)
9430 RETURN
9900 FOR I=0 TO 999:POKE CRT+I,49:NEXT
9910 GET T$:IF T$="" THEN 9910
9920 RETURN
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$(" .{+}",ZC,1); "{left}";:ZC=3-ZC:
    ZT=TI+15

```

```
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT ",{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}"
(C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100
```



BAT (Uses joystick)

By Chris Nadovich

Life isn't easy for a bat, as you'll find out. In this game you will lead the life of a bat; you will learn to avoid bouncing off walls, make quick decisions about good-tasting and bad-tasting bugs, and avoid starving to death. As a bat, you must fly around and find insects to eat. Some of the insects are tasty and will help you stay alive. But the so-called "dirt bug" is worthless junk food, and "lint" actually reduces your current calories by half.

The insects have the following values:

Blue	Elephant bug	25 calories
Green	Buffet beetle	10 calories
Yellow	Feast fly	Doubles existing calories
Red	Lint	Halves existing calories
Gray	Dirt bug	Worthless

Use the joystick to control the bat. Sound will accompany every movement of the joystick.

Remember, you're always burning calories even if you're not flying. You get 50 calories when the game begins. To liven things up a bit, your radar doesn't work very well, so you will often fly into walls in the dark and bounce off them. There are some techniques that successful players of BAT soon learn. The most important technique is to avoid bumping into the sides and top of the screen, due to the wild "bounce." Another useful hint is to try to hover by applying a small amount of upward force, but very little sideways motion. The game ends when all of your calories have been used up. On the right side of the screen, you will see the time that you existed as a bat. Although BAT is frustrating to beginning players, those with enough patience to learn the special skills needed to control the bat find this a challenging game.

```
0 PG$="^B^A^T^":AU$="BY^C.^T.^NADOVICH":JF=1
1 REM BAT -- BY C.T.NADOVICH
4 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 9/6/83 GF
90 GOTO 62000
100 GOSUB 5000
103 FOR I=0 TO 28:POKE SID+I,0:NEXT:POKE SID+6,240:POKE SID+4,17
104 VL=SID+24
130 DIM T(29):FOR I=0 TO 29:T(I)=INT(32*(2↑(I/12))-1.5):NEXT I
140 Q8=SID+1:PRINT "{clr}":POKE VL,15:POKE Q8,0
150 D(1)=26:D(2)=18:D(3)=25:D(4)=42:D(6)=41:D(7)=58:D(8)=50:D(9)=57
200 G=9.8:REM GRAVITATIONAL ACCEL.
210 U=.3:REM FRICTION COEFF.
220 A=20:REM THRUST ACCEL.
250 U=1-U:G=G/(9.8*4):A=A/(9.8*4)
310 FOR I=1 TO 9
320 VX(I)=VX(I)*A:VY(I)=VY(I)*A:NEXT I
499 REM START OF GAME
```

```
500 RESTORE:GF=.142:TP=50
510 TQ=TI:ZW=32:S0=1024
520 T=PEEK(VIC+17):POKE VIC+17,0
530 POKE VIC+33,1:PRINT "{clr}":POKE VIC+33,0:POKE VIC+17,T
560 PRINT "home cyn rvs-on 6°space rvs-off wht}CALORIES:{4°space
cyn rvs-on}~~~{rvs-off wht}TIME{7°space cyn rvs-on 7°space}";
610 FOR I=1 TO 23:PRINT "{left down}~":"NEXT
620 FOR I=1 TO 39:PRINT "{2°left}~":"NEXT
630 FOR I=1 TO 24:PRINT "{up left}~":"NEXT
640 PRINT "{home wht}":GOSUB 2000
700 X=20:Y=12:VY=0:VX=0:TI$="000000"
1000 K=S0+INT(X+.5)+WD*INT(Y+.5)
1010 R=PEEK(K):TP=TP-GF*(TP/50+1)
1020 IF Q9=1 THEN N=255:Q9=0:GOTO 1040
1030 IF Q9=0 THEN N=127:Q9=1:GOTO 1040
1040 IF R=32 GOTO 1150
1090 IF R=214 THEN TP=TP+25:POKE Q8,T(21):GOTO 1140
1100 IF R=88 THEN TP=TP+10:POKE Q8,T(24):GOTO 1140
1110 IF R=83 THEN TP=TP*2:POKE Q8,T(29):GOTO 1140
1120 IF R=42 THEN POKE Q8,T(1):TP=TP/2
1140 POKE K-CRT+CM,1:GOSUB 4000
1150 POKE K,N
1180 GET Q$:IF Q$="Q" THEN PRINT "{clr}":END
1185 C=FNJ(0)
1190 TP=TP-GF*A
1200 IF C<>5 THEN POKE Q8,T(SD(C)):VX=VX+VX(C):VY=VY+VY(C)
1600 VY=VY+G:NY=Y+VY:NX=X+VX:POKE Q8,0
1700 IF NX>38 THEN NX=38:VY=U*VY:VX=-VX*U
1750 IF NX<1 THEN NX=1:VY=U*VY:VX=-VX*U
1800 IF NY>22 THEN NY=22:VX=U*VX:VY=-VY*U
1850 IF NY<1 THEN NY=1:VX=U*VX:VY=-VY*U
1900 Y=NY:X=NX
1910 IF TP<0 GOTO 3000
1920 N=32:POKE K,N
1930 IF Q9=1 GOTO 1990
1940 POKE Q8,0
1950 PRINT "{home wht}";TAB(15);INT(TP);"{left}~":";
      PRINT "{home}";TAB(27);
1960 PRINT MID$(TI$,3,2) ":" MID$(TI$,5,2)
1990 GOTO 1000
2000 FOR I=1 TO 5
2020 FOR J=1 TO BN(I)
2030 BX=2+INT(36*RND(1))
2040 BY=2+INT(21*RND(1))
2050 POKE QL,BY+1:PRINT TAB(BX);"{up}";BG$(I)
2060 NEXT
2070 NEXT:PRINT "{home wht}"
2999 RETURN
```

```

3000 REM DUST TO DUST
3001 PRINT "{home 10°down}";TAB(14);"{rvs-on}YOU ARE DEAD{rvs-off}":
    RESTORE
3005 FOR K=1 TO 11
3006 READ Q5,Q4:Q4=200*Q4
3007 POKE Q8,T(Q5)
3008 FOR J=1 TO Q4:NEXT
3009 POKE Q8,0
3010 NEXT
3012 DATA 5,4,5,3,5,1,5,4,8,3,7,1,7,3,5,1,5,3,4,1,5,4
3030 PRINT "{clr}":GOSUB 3100:IF YN THEN 500
3050 POKE VL,0:PRINT "{clr wht}":END
3100 NO=0
3110 TM=0:SW=1:PRINT "{wht}TRY AGAIN?{shift-space cyn}YES{up}"
3120 IF TI<TM THEN 3150
3130 PRINT TAB(11+NO*4);MID$("{wht cyn}",SW,1);
    MID$("YESNO{shift-space}",3*NO+1,3);"{up}"
3140 SW=3-SW:TM=TI+15
3150 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
3160 IF JB AND JD=15 THEN YN=(NO=0):GOTO 3200
3170 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 3110
3180 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 3110
3190 GOTO 3120
3200 PRINT TAB(11+NO*4);"{wht}";MID$("YESNO{shift-space}",3*NO+1,3):
    RETURN
4000 BX=2+INT(36*RND(1))
4010 BY=2+INT(21*RND(1))
4040 NB=INT(RND(1)*5+1)
4060 POKE QL,BY+1:PRINT TAB(BX);"{up}";BG$(NB)
4999 RETURN
5000 READ T$:IF T$<>"£" THEN 5000
5010 DIM SD(9),VX(9),VY(9),JD(15),BG(5),BG$(5),BN(5)
5020 FOR I=0 TO 15:READ JD(I):NEXT:DATA £,0,0,0,0,0,3,9,6,0,1,7,4,0,
    2,8,5
5030 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
    16)=0
5040 FOR I=1 TO 9:READ SD(I),VX(I),VY(I):NEXT I
5060 DATA 14,-1,1, 17,0,1, 16,1,1, 10,-1,0, 0,0,0
5070 DATA 12,1,0, 7,-1,-1, 5,0,-1, 9,1,-1
5090 FOR I=1 TO 5
5100 READ Z$,BN(I):BG$(I)=Z$":PRINT "{home blk}";MID$(Z$,2):
    BG(I)=PEEK(CRT):NEXT I
5110 DATA "{grn rvs-on V}",5,"{blu X}",4,"{red}*",5,"{yel S}",4,
    "{pur A}",3
5120 PRINT "{wht}":RETURN
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$":IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);"{left}":ZC=3-ZC:
    ZT=TI+15

```

```

60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "^{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}"
(C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MSS,
Q)
62100 Q=Q+1:IF Q>LEN(MSS) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```

RESCUE (Uses joystick)

By Nick Jackiw



One of the ships in your space fleet has crashed on an alien planet, and you must rescue its survivors from the very unfriendly inhabitants. Your mother ship cannot land on the uneven terrain, so you must ferry the survivors back to the ship in your shuttle. This means landing the shuttle safely on the landing pad, picking up a survivor, and carrying him or her back to the mother ship. Survivors can only be carried one at a time, and they are found in groups of five. The more survivors you rescue, the harder the aliens make it for you to complete your mission.

There are three levels of difficulty: easy (1), average (2), and hard (3). At the "easy" level, you start off with five shuttles, and the aliens have one row of orbiting attack craft. Whenever you are able to rescue an entire group of five survivors, you get a bonus. The aliens keep adding rows of attackers (up to the maximum of three).

But that's not all; they fire weapons that produce permanent "space flak," which you must dodge unless you want to crash.

When you select "average," you get three shuttles, and the enemy starts off with two rows of attackers. The most difficult level gives you only one shuttle, and the aliens begin with three rows of attack craft. There's one more problem: after you rescue a group of five when the enemy has three rows of attackers, they reduce the size of your landing pad!

Push the joystick down to launch your shuttle from the mother ship, and control the shuttle with the joystick. All missions begin with 3000 gallons of fuel in the mother ship. Points are scored for each person you save, and as the missions get harder, the survivors are worth more points. The game is over either when you have no more shuttles or when your ship runs out of fuel.

```
0 PG$="^R^S^C^U^E":AU$="BY^NICK^JACKIW":JF=1
2 :
5 REM COPYRIGHT (C) 1983 THE CODE WORKS
6 REM BOX 6905, SANTA BARBARA, CA 93160
7 :
10 REM AS OF 9/6/83 GF
90 GOTO 62000
100 DIM JD(15):FOR I=0 TO 15:READ JD(I):NEXT
101 DATA 0,0,0,0,0,6,6,6,0,4,4,4,0,2,8,0
102 DEF FNJ(X)=JD(PEEK(JS+X) AND 15):DEF FNB(X)=(PEEK(JS+X) AND
16)=0
103 GOSUB 1000:OF=CM-CRT:CP=7:CR=CRT-1:MV=88
104 POKE SID+5,1:POKE SID+6,255:POKE SID+24,15:POKE SID+4,33
110 MS=1:F=3000:S=0:L=SJ-1:LV=LF:HS=0:RU=0:BN=50
112 FOR I=0 TO 4:FOR J=0 TO 4:X(I,J)=0:NEXT:NEXT
114 FOR J=1 TO L+1:ON J GOSUB 590,600,610:NEXT J
115 PRINT "{clr}";
116 FOR CV=1 TO 23:PRINT "{wht 39°space}":NEXT
117 GOSUB 280
120 POKE SID+24,15:POKE SID+1,0
125 DNS="{19°down}"
130 PRINT DNS " {pur 6°T M}"
```

```
140 PRINT "{yel 5°Q pur}\u2022{shift-space 4°T M 5°space N 8°T up N  
12°T left inst T down}"  
150 PRINT "{yel 12°space pur M yel 3°£ pur N}"  
155 GOSUB 800  
160 PRINT "{home}":X=1:US=87:Y=1:V=2:AD=-1:IF RU=0 THEN RU=1:  
GOSUB 580  
170 POKE CR+WD+X,US:POKE CR+WD+X+OF,CP:GOSUB 280  
175 IF F<=0 THEN GOSUB 290:GOTO 640  
180 D$=RIGHT$(STR$(FNJ(0)),1):IF D$="0" THEN 190  
183 Y=2:SS=CR+X+WD:POKE SS,MV  
185 POKE CR+X+WD*Y,US:POKE CR+X+WD*Y+OF,CP:POKE SID+1,99:GOTO 320  
190 GET Q$:IF Q$="Q" THEN PRINT "{clr wht}":POKE SID+1,0:  
POKE SID+15,0:END  
195 GOSUB 230  
200 F=F-2:IF F=0 THEN GOSUB 290:GOTO 640  
210 POKE CR+WD+X,32:X=X+Y:IF X=40 OR X=1 THEN Y=-Y  
220 GOTO 170  
230 FOR I=0 TO L-1:X(I,2)=X(I,2)+1:IF X(I,2)<X(I,0) THEN 260  
240 X(I,2)=0:X$(I)=RIGHT$(X$(I),1)+LEFT$(X$(I),LEN(X$(I))-1):  
PRINT "{home}";  
250 PRINT LEFT$("red grn blu",I+1)LEFT$(DN$,X(I,1))X$(I) " {home  
wht}";  
260 NEXT I:ER=0:PP=PEEK(PP):IF PP=86 OR PP=90 OR PP=105 THEN ER=1  
270 RETURN  
280 PRINT "{home wht}FUEL:" MID$(STR$(F),2) " ^~"  
285 PRINT "{home wht}" TAB(12) " ^~SCORE:";MID$(STR$(S),2)  
" {17°space}":RETURN  
290 POKE SID+24,15:POKE SID+1,0:FOR J=1 TO 6  
295 POKE SID+1,246:PRINT "{home rvs-on wht}OUT_OF_FUEL!":  
FOR I=1 TO 99:NEXT I  
300 POKE SID+1,165:PRINT "{home wht}OUT_OF_FUEL!":FOR I=1 TO 99:  
NEXT I,J  
305 POKE SID+1,0:POKE SID+24,0:FOR I=1 TO 200:NEXT I:LV=0:RETURN  
320 U=0:Z=FNJ(0)  
321 POKE SID+1,81  
330 D$=C$:POKE SID+1,0  
335 GET Q$:IF Q$="Q" THEN PRINT "{clr wht}":POKE SID+1,0:  
POKE SID+15,0::END  
340 IF Z=4 THEN F=F-10:GOSUB 540:ON GT GOTO 400,420  
350 IF Z=6 THEN F=F-10:GOSUB 560:ON GT GOTO 400,420  
360 IF FNB(0)<>0 AND Y>3 THEN F=F-50:Y=Y+AD  
370 POKE CR+X+Y*WD,32:Y=Y-AD:A=0:U=1:P=PEEK(CR+X+Y*WD):GOTO 420  
380 IF RND(1)*30>=L THEN 400  
381 KY=INT(RND(1)*15+3):KX=INT(RND(1)*39)+1  
382 IF KY=9 OR KY=12 OR KY=16 OR KX=X THEN 400  
385 K=CRT+KY*WD+KX  
390 IF PEEK(K)=32 OR PEEK(K)=96 THEN POKE K,42:POKE K+OF,8  
400 GOSUB 230:IF ER THEN 635
```

```

410 F=F-1:GOSUB 280:IF F>0 THEN 320
411 GOSUB 290:F=0:GOTO 640
420 IF P=104 THEN GOSUB 470:GOTO 320
430 IF P=MV THEN GOSUB 900:GOSUB 520:POKE SID+1,0:GOTO 160
440 IF P<>US AND P<>32 AND P<>96 THEN 635
450 PP=CR+X+Y*WD:POKE PP,US:POKE PP+OF,CP:ON U+1 GOTO 370,380
470 POKE CR+X+Y*WD,87:HS=HS+1:POKE SID+1,120
480 POKE CR+21*WD+HS,32
485 JL=32:JM=190:FOR JK=CR+21*WD+HS+WD TO CR+X+Y*WD
486 POKE JK-1,JL:JL=PEEK(JK):POKE JK,81:POKE SID+1,JM:
FOR J=1 TO 30:NEXT
487 JM=JM-10:POKE SID+1,0:NEXT:US=81
488 POKE CR+X+Y*WD,US:POKE CR+X+Y*WD+OF,CP
490 IF FNJ(0)=0 THEN GOSUB 230:GOSUB 850:GOTO 490
495 POKE SID+1,100:FOR J=1 TO 40:NEXT
500 POKE CR+X+Y*WD,104:Y=Y-2:AD=-AD:POKE SID+1,0:RETURN
520 POKE SS,32:IF DF AND US=81 THEN AP=1
525 IF DF=0 THEN S=S+(MS+1)*10
530 IF HS=5 AND AP=0 THEN S=S+BN:GOSUB 955
535 IF HS=5 THEN HS=0:AP=0:MS=MS+1:RU=0:PRINT "{home}";DN$;
"2°down yel 5°Q wht":GOSUB 800
537 GOSUB 280:FOR I=1 TO 750:NEXT I:RETURN
540 IF X>1 THEN POKE CR+X+Y*WD,32:X=X-1:P=PEEK(CR+X+Y*WD):GT=2:
RETURN
550 GT=1:RETURN
560 IF X<WD THEN POKE CR+X+Y*WD,32:X=X+1:P=PEEK(CR+X+Y*WD):GT=2:
RETURN
570 GT=1:RETURN
580 L=L+1:ON L GOSUB 590,600,610,620
590 X$(0)="{£} ^{3°right} ^{£} ^{3°right} ^{£} ^{3°right} ^{£} "
"^{3°right} ^{£} ^{3°right} ^{£} ^{3°right} ^{£} ^{3°right} ^{£} "
:X(0,0)=4:X(0,1)=12:RETURN
600 X$(1)="{Z} ^{right} ^{Z} ^{right} ^{Z} ^{right} ^{Z} ^{right} "
"^{Z} ^{right} ^{Z} ^{right} ^{Z} ^{right} ^{Z} ^{right} "
:X(1,0)=3:X(1,1)=16:
RETURN
610 X$(2)="-{right} -{V} ^{right} -{V} ^{right} -{V} ^{right} "
"-{V} ^{right} -{V} ^{right} -{V} ^{right} "
:X(2,0)=2:X(2,1)=9:RETURN
620 L=L-1:POKE CR+22*WD+16,100:POKE CR+22*WD+14,100:RETURN
635 Q=CR+X+INT(Y)*WD:Z=PEEK(Q):POKE Q,160:
PRINT "{home}CRASH!{5°space}"
636 FOR J=1 TO 4:PRINT "{home rvs-on}CRASH!":FOR J1=1 TO 20:
POKE SID+1,255:POKE SID+1,0
637 NEXT J1:PRINT "{home}CRASH!":FOR J1=1 TO 99:NEXT:NEXT:
FOR J=1 TO 500:NEXT
638 POKE Q,Z:DF=1:GOSUB 520:DF=0
640 LV=LV-1:IF LV>0 THEN POKE SS,32:GOTO 155
650 IF S>BS THEN BS=S:BF=F
660 IF MS>BM THEN BM=MS
670 IF F<0 THEN F=0

```

```

675 MT=MT+MS:FT=FT+F:NT=NT+1:TS=TS+S:AVG=INT(TS/NT):FA=INT(FT/NT)
678 AVG=INT(TS/NT):FA=INT(FT/NT):MA=MT/NT:MA=.1*(INT(MA*10))
680 DF$="EASY":IF SJ>1 THEN DF$="AVERAGE":IF SJ=3 THEN DF$="HARD"
685 PRINT "{clr 2°down rvs-on 8°right wht}MISSION CONTROL REPORT"
687 PRINT "{3°down 4°space}LEVEL:~" DF$
689 PRINT "{down}" TAB(27) "# OF"
690 PRINT TAB(18) "SCORE";TAB(25);"MISSIONS"
692 PRINT "{down 4°space}THIS GAME:" TAB(19) S;TAB(27) MS
700 IF NT<2 THEN 730
710 PRINT "{down 4°space}BEST SO FAR:" TAB(19) BS;TAB(27) BM
720 PRINT "{down 4°space}AVERAGE:" TAB(19) AV;TAB(27) MA
730 GOSUB 9100
740 IF YN=0 THEN POKE SID+24,0:PRINT "{clr wht}":END
750 PRINT "{clr wht}":GOTO 110
800 PRINT "{home wht}";DNS;"{4°down}MISSION:";MS;"~~~GAME:";NT+1;
    "~SHUTTLES LEFT:";LV-1;
810 RETURN
850 F=F-1:GOSUB 280:FOR I=1 TO 80:NEXT I:RETURN
900 REM SOUND
910 FOR J=100 TO 30 STEP -3:POKE SID+1,J:NEXT:POKE SID+1,0:RETURN
950 REM BONUS
955 PRINT "{home}" TAB(25) "{rvs-on}BONUS!!!{rvs-off}"
960 POKE SID+1,150:FOR J=1 TO 300:NEXT:POKE SID+1,100:
    FOR J=1 TO 300:NEXT
965 POKE SID+1,50:FOR J=1 TO 300:NEXT:POKE SID+1,0:RETURN
1000 PRINT "{clr 4°down}":GOSUB 7000
1030 IF G=1 THEN LF=5:SJ=1:RETURN
1040 IF G=2 THEN LF=3:SJ=2:RETURN
1050 IF G=3 THEN LF=1:SJ=3:RETURN
7000 PRINT "{clr wht 2°down}PICK A SKILL LEVEL:{down cyn}":
7010 FOR I=1 TO 3:PRINT I,:NEXT I:PRINT:G=1:T=1:TM=0
7020 X=G:IF TI>TM THEN PRINT TAB(3*G-3);MID$("{wht cyn}",T,1);
    "{up}";G:T=3-T:TM=TI+10
7030 IF FNJ(0)=4 THEN IF G>1 THEN G=G-1
7040 IF FNJ(0)=6 THEN IF G<3 THEN G=G+1
7050 IF X<>G THEN PRINT TAB(3*X-3);"{cyn up}";X:T=1:TM=0
7060 IF FNB(0)=0 THEN 7020
7070 RETURN
9100 NO=0:PRINT "{2°down}"
9110 TM=0:SW=1
9115 PRINT "{wht}PLAY AGAIN?~{cyn}YES~NO{up}"
9120 IF TI<TM THEN 9150
9130 PRINT TAB(12+NO*4);MID$("{wht cyn}",SW,1);
    MID$("YESNO{shift-space}",3*NO+1,3);"{up}"
9140 SW=3-SW:TM=TI+15
9150 JD=PEEK(JS) AND 31:JB=(JD<16):JD=JD AND 15
9160 IF JB AND JD=15 THEN YN=(NO=0):RETURN
9170 IF (JD AND 8)=0 AND NO=0 THEN NO=1:GOTO 9115
9180 IF (JD AND 4)=0 AND NO=1 THEN NO=0:GOTO 9115

```

```

9190 GOTO 9120
60000 IN$="~":"ZT=TI:ZC=2:ZD$=CHR$(20)
60010 GET Z$:IF Z$<>"" THEN 60070
60020 IF ZT<=TI THEN PRINT MID$("~{+}",ZC,1);"{left}";:ZC=3-ZC:
ZT=TI+15
60030 GOTO 60010
60070 Z=ASC(Z$):ZL=LEN(IN$):IF (Z AND 127)<32 THEN PRINT "~{left}";:
GOTO 60110
60090 IF ZL>QI THEN 60010
60100 IN$=IN$+Z$:PRINT Z$;ZD$;Z$;
60110 IF Z=13 THEN IN$=MID$(IN$,2):PRINT CR$;:RETURN
60120 IF Z=20 AND ZL>1 THEN IN$=LEFT$(IN$,ZL-1):PRINT "{left}";:
GOTO 60010
60130 IF Z=141 THEN Z$=CHR$(-20*(ZL>1)):FOR Z=2 TO ZL:PRINT Z$;:NEXT:
GOTO 60000
60140 GOTO 60010
61000 CRT=1024:VIC=53248:WD=40:CR$=CHR$(13):SID=54272:JS=56320:
CM=55296
61010 QL=214:QI=254:RETURN
62000 GOSUB 61000:POKE VIC+32,0:POKE VIC+33,0:LN=LEN(PG$):
TB=(40-LN)/2
62010 PRINT "{clr 4°down yel rvs-on}" TAB(TB)LEFT$("{21°space}",LN)
62020 PRINT "{rvs-on}" TAB(TB)PG$:PRINT "{rvs-on}"
TAB(TB)LEFT$("{21°space}",LN)
62030 PRINT "{2°down}" TAB((40-LEN(AU$))/2)AU$
62040 MS$="HIT^RETURN^TO^BEGIN":IF JF THEN
MS$="HIT^JOYSTICK^BUTTON^TO^BEGIN"
62050 TB=((40-LEN(MS$))/2)
62060 PRINT "{4°down cyn}" TAB(TB)MS$:PRINT "{7°down cyn 9°space}
(C)^1983^THE^CODE^WORKS"
62070 GET IN$:IF IN$<>"" THEN 62120
62080 IF (PEEK(JS) AND 16)=0 THEN 62120
62090 PRINT "{home 14°down}" TAB(TB)LEFT$("{pur cyn}",K+1)LEFT$(MS$,
Q)
62100 Q=Q+1:IF Q>LEN(MS$) THEN Q=0:K=1-K
62110 GOTO 62070
62120 CLR:PRINT "{clr wht}":GOSUB 61000:GOTO 100

```


COMMODORE 64TM

Fun and Games

Jeffries
Fisher
Sawyer

Key in to the entertainment potential of your Commodore 64TM with this test-proven collection of 35 games and puzzles!

Enhanced with humor and imagination, **C-64TM Fun & Games** gives you easy-to-understand instructions and complete BASIC program listings for each game.

You will soon be playing games such as Bat, Everest, and Yahtzee, and enjoying others that have been designed to take full advantage of your C-64TM's range of color graphics and sound capabilities.

The number and quality of the games presented in **C-64TM Fun & Games** make this collection an exceptional value.

C-64TM Fun & Games will bring out the playfulness in you and your C-64!

Commodore 64 is a trademark of Commodore Business Machines, Inc.



ISBN 0-88134-116-9